HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



NGUYỄN PHƯƠNG ANH

ỨNG DỤNG MÔ HÌNH THỊ GIÁC-NGÔN NGỮ LỚN VÀO BÀI TOÁN NHẬN DẠNG VÀ TRÍCH XUẤT THÔNG TIN TỪ ẢNH TÀI LIỆU TIẾNG VIỆT

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

HÀ NỘI - NĂM 2025

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



NGUYỄN PHƯƠNG ANH

ỨNG DỤNG MÔ HÌNH THỊ GIÁC-NGÔN NGỮ LỚN VÀO BÀI TOÁN NHẬN DẠNG VÀ TRÍCH XUẤT THÔNG TIN TỪ ẢNH TÀI LIỆU TIẾNG VIỆT

Chuyên ngành: Khoa học máy tính
Mã số: 8.48.01.01

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT
(*Theo định hướng ứng dụng*)

NGƯỜI HƯỚNG DẪN KHOA HỌC :
TS. Nguyễn Đình Hóa

HÀ NỘI - NĂM 2025

LỜI CAM ĐOAN

Tôi xin cam đoan đề án thạc sĩ "Ứng dụng mô hình thị giác–ngôn ngữ lớn vào bài toán nhận dạng và trích xuất thông tin từ ảnh tài liệu tiếng Việt" của tôi được thực hiện dưới sự hướng dẫn của giáo viên hướng dẫn là TS. Nguyễn Đình Hóa.

Tôi xin khẳng định rằng toàn bộ nội dung trong đề án này đều được trích dẫn đầy đủ nguồn tham khảo ở phần cuối tài liệu. Các dữ liệu, kết quả và phân tích được trình bày trong đề án là trung thực và chưa từng được công bố trong bất kỳ nghiên cứu hay ấn phẩm nào khác. Nếu có phát sinh sai sót liên quan đến việc sao chép tài liệu hoặc sử dụng kết quả nghiên cứu của người khác mà không dẫn nguồn đúng quy định, tôi hoàn toàn chịu trách nhiệm về mọi hậu quả liên quan đến đề án này.

Học viên thực hiện



Nguyễn Phương Anh

LỜI CẢM ƠN

Lời đầu tiên, tôi xin trân trọng gửi lời cảm ơn sâu sắc nhất tới TS. Nguyễn Đình Hóa. Thầy không chỉ định hướng đề tài mà còn tận tình hỗ trợ, động viên và góp ý chi tiết trong suốt quá trình tôi thực hiện đề án thạc sĩ. Sự hướng dẫn tận tâm của thầy là nền tảng quan trọng giúp tôi hoàn thành tốt nhiệm vụ nghiên cứu này.

Tôi cũng xin gửi lời cảm ơn chân thành đến toàn thể các thầy, cô của Học viện Công nghệ Bưu chính Viễn thông. Trong suốt thời gian học tập tại Học viện, tôi đã nhận được rất nhiều kiến thức quý báu cùng sự dìu dắt tận tình. Những bài giảng, những chia sẻ và sự hỗ trợ của các thầy cô không chỉ giúp tôi củng cố nền tảng học thuật mà còn tạo điều kiện thuận lợi để tôi phát triển tư duy nghiên cứu và hoàn thiện đề án.

Học viên thực hiện



Nguyễn Phương Anh

MỤC LỤC

LỜI CAM ĐOAN 3
LỜI CẢM ƠN 4
MỤC LỤC 5
DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT 7
DANH SÁCH BẢNG 8
DANH SÁCH HÌNH VẼ 9
CHƯƠNG 1: TỔNG QUAN VỀ BÀI TOÁN 4
1.1. Giới thiệu chung về bài toán OCR 4
1.2. Đặc thù tiếng Việt trong bài toán OCR 5
1.2.1. Hệ thống chữ viết và dấu tiếng Việt 5
1.2.2. Bố cục tài liệu và đa dạng nguồn dữ liệu 5
1.2.3. Chữ in và chữ viết tay 6
1.3. Các hướng tiếp cận OCR hiện nay 6
1.3.1. Pipeline OCR truyền thống 6
1.3.2. OCR dựa trên học sâu 9
1.4. Các nghiên cứu về OCR tiếng Việt 9
1.4.1. Nhận dạng chữ in và văn bản in trong tài liệu 10
1.4.2. Nhận dạng chữ trong ảnh cảnh 10
1.4.3. Nhận dạng chữ viết tay tiếng Việt 11
1.5. Xu hướng mô hình thị giác–ngôn ngữ và OCR tài liệu tiếng Việt 12
1.6. Đánh giá chung và khoảng trống nghiên cứu 14
1.7. Kết luận chương 16
CHƯƠNG 2: PHƯƠNG PHÁP ĐỀ XUẤT 17
2.1. Định hướng và mục tiêu của phương pháp 17
2.2. Cơ sở lý thuyết 18
2.2.1. Cơ sở lý thuyết về mô hình thị giác–ngôn ngữ (VLM) 18
2.2.2. Kiến trúc mô hình Qwen 2.5-VL 7B 22
Điểm nổi bật 25
2.2.3. Cơ sở lý thuyết về fine-tuning và LoRA 27
2.3. Thiết kế dữ liệu huấn luyện cho Qwen 2.5-VL 30
2.3.1. Loại dữ liệu 30
2.3.2. Tiền xử lý và chuẩn hóa 30
2.3.3. Định dạng dữ liệu theo phong cách hội thoại 31
2.4. Pipeline tinh chỉnh Qwen 2.5-VL bằng LoRA 31
2.4.1. Các bước tổng quát 31

2.4.2. Cấu hình huấn luyện cụ thể....33
2.5. Quy trình suy luận (inference) cho OCR....33
2.6. Thiết lập benchmark và tiêu chí đánh giá....34
2.7. Kết luận chương....35
CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ....36
3.1. Bộ dữ liệu thực nghiệm....36
3.2. Độ đo đánh giá (Metrics)....40
3.2.1. CER – Character Error Rate....40
3.2.2. WER – Word Error Rate....41
3.3. Quy trình tinh chỉnh mô hình....42
3.4. Kết quả thực nghiệm....47
3.5. So sánh mở rộng với các mô hình OCR truyền thống....51
3.6. Xây dựng ứng dụng....53
3.7. Kết luận chương....56
KẾT LUẬN....58
TÀI LIỆU THAM KHẢO....59

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Ký hiệu / Viết tắt	Tiếng Anh	Nghĩa tiếng Việt
OCR	Optical Character Recognition	Nhận dạng ký tự quang học
VLM	Vision–Language Model	Mô hình thị giác–ngôn ngữ
SFT	Supervised Fine-Tuning	Tinh chỉnh có giám sát
LoRA	Low-Rank Adaptation	Tinh chỉnh hạng thấp
ViT	Vision Transformer	Bộ mã hóa ảnh dạng Transformer
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập
RNN	Recurrent Neural Network	Mạng nơ-ron hồi tiếp
LSTM	Long Short-Term Memory	Mạng hồi tiếp LSTM
CTC	Connectionist Temporal Classification	Thuật toán ánh xạ chuỗi không căn chỉnh
CRNN	Convolutional Recurrent Neural Network	Mạng CNN kết hợp RNN
CER	Character Error Rate	Tỉ lệ lỗi ký tự
WER	Word Error Rate	Tỉ lệ lỗi từ
VQA	Visual Question Answering	Hỏi–đáp dựa trên ảnh
PEFT	Parameter-Efficient Fine-Tuning	Kỹ thuật tinh chỉnh tối ưu tham số
EAST	Efficient and Accurate Scene Text Detector	Mô hình phát hiện văn bản trong ảnh

DANH SÁCH BẢNG

Bảng 3.1: Cấu hình huấn luyện mô hình Qwen 2.5-VL 7B + LoRA.....42

Bảng 3.2: Kết quả thực nghiệm so sánh baseline.....45

Bảng 3.3: Kết quả thực nghiệm theo loại dữ liệu.....46

Bảng 3.4: Hiệu suất OCR theo độ dài chuỗi văn bản.....47

Bảng 3.5:Kết quả thực nghiệm so sánh thêm các baseline OCR truyền thống.....52

DANH SÁCH HÌNH VẼ

Hình 1.1: Pipeline OCR truyền thống.....7

Hình 2.1: Kiến trúc chung của mô hình VLM..... 18

Hình 2.2: Kiến trúc mô hình Qwen 2.5-VL.....22

Hình 2.3: Quá trình fine-tuning mô hình lớn.....27

Hình 2.4: Cơ chế LoRA trong quá trình tinh chỉnh..... 28

Hình 3.1: Dữ liệu UIT-HWDB dạng dòng.....36

Hình 3.2: Dữ liệu UIT-HWDB dạng đoạn.....36

Hình 3.3: Dữ liệu Vietnamese_original..... 36

Hình 3.4: Dữ liệu Vietnamese Handwriting OCR..... 37

Hình 3.5: Biểu đồ phân bố dữ liệu train.....38

Hình 3.6: Biểu đồ xu hướng loss..... 44

Hình 3.7: Biểu đồ hiệu xuất theo độ dài chuỗi văn bản.....47

Hình 3.8: Giao diện ứng dụng.....48

Hình 3.9: Thử nghiệm với hình ảnh đa ngôn ngữ.....49

Hình 3.10: Thử nghiệm chữ in.....49

Hình 3.11: Thử nghiệm chữ viết tay khó..... 49

Hình 3.12: Thử nghiệm chữ viết tay..... 50

Hình 3.13: Thử nghiệm hỏi đáp thông tin.....50

MỞ ĐẦU

Trong bối cảnh chuyển đổi số đang diễn ra sâu rộng tại Việt Nam, nhu cầu số hóa tài liệu giấy và tự động hóa quy trình xử lý văn bản đang trở thành một yêu cầu cấp thiết đối với cả khu vực công lẫn tư. Các loại tài liệu như hóa đơn, chứng từ, phiếu giao dịch, biểu mẫu hành chính, giấy tờ cá nhân hay hồ sơ nghiệp vụ vẫn chiếm tỷ trọng lớn trong hoạt động của doanh nghiệp và cơ quan nhà nước. Việc xử lý thủ công không chỉ tốn thời gian mà còn dễ xảy ra sai sót, gây ảnh hưởng đến năng suất và chất lượng quản lý dữ liệu. Báo cáo Chuyển đổi số Quốc gia năm 2022 cho thấy hơn 90% doanh nghiệp đang ưu tiên đầu tư vào hệ thống số hóa dữ liệu, trong đó OCR được xem là công nghệ nền tảng để tự động hóa đầu vào thông tin. Tuy nhiên, để triển khai OCR hiệu quả, chất lượng nhận dạng phải đạt mức đủ cao để đảm bảo không gây lỗi trong quy trình vận hành.

Dù đã có lịch sử phát triển lâu dài và được ứng dụng trong nhiều hệ thống, các phương pháp OCR truyền thống vẫn tồn tại những hạn chế nhất định, đặc biệt đối với tiếng Việt. Các hệ thống như Tesseract, PaddleOCR hay các mô hình CRNN – Transformer trong pipeline detection–recognition hoạt động tốt với ngôn ngữ có cấu trúc đơn giản, nhưng gặp nhiều khó khăn với tiếng Việt do đặc thù hệ thống dấu thanh, sự đa dạng về font chữ và độ biến thiên lớn của tài liệu thực tế. Ảnh chụp từ thiết bị di động, tài liệu mờ, lệch hoặc có bố cục bất thường đều làm tăng tỷ lệ lỗi nhận dạng. Cùng với đó, chữ viết tay tiếng Việt – vốn chưa có bộ quy chuẩn rõ ràng – tiếp tục là thách thức lớn đối với cả mô hình truyền thống và nhiều mô hình OCR hiện đại.

Trong bối cảnh hạn chế của các phương pháp OCR truyền thống, sự xuất hiện của các mô hình thị giác–ngôn ngữ lớn (Vision-Language Models – VLMs) mở ra một hướng tiếp cận mới đầy tiềm năng. Không chỉ dừng lại ở việc trích xuất ký tự, VLM có khả năng hiểu nội dung trực tiếp từ ảnh tài liệu và sinh văn bản tương ứng thông qua cơ chế học đa mô thức. Những mô hình như LayoutLM [24], Donut [25], Qwen

2.5 VL đã chứng minh hiệu quả vượt trội trên nhiều tác vụ liên quan đến tài liệu, đặc biệt trong môi trường hình ảnh phức tạp. Khả năng xử lý đồng thời thông tin hình ảnh và ngôn ngữ giúp VLM nắm bắt ngữ cảnh toàn cục của trang tài liệu thay vì chỉ các vùng văn bản riêng lẻ.

Trong số đó, Qwen 2.5-VL – mô hình thị giác–ngôn ngữ thế hệ mới do Alibaba phát triển – đã cho thấy năng lực vượt trội trong việc hiểu tài liệu và sinh văn bản trực tiếp mà không cần pipeline OCR tách biệt. Nhờ kiến trúc tích hợp thị giác và ngôn ngữ, mô hình có thể xử lý các tài liệu phức tạp như ảnh mờ, scan lệch, bố cục nhiều cột, bảng biểu hoặc chữ viết tay. Điều này đặc biệt quan trọng trong bối cảnh tiếng Việt, nơi dữ liệu thực tế thường có chất lượng không đồng đều. Tuy nhiên, dù có tiềm năng lớn, Qwen 2.5-VL vẫn cần được tinh chỉnh (fine-tuning) để phù hợp hơn với ngôn ngữ và phân bố dữ liệu tiếng Việt.

Mặc dù VLM đã trở thành xu hướng mới trong xử lý tài liệu, hiện nay vẫn thiếu các nghiên cứu đánh giá có hệ thống về hiệu quả của chúng đối với tiếng Việt. Các bộ dữ liệu OCR tiếng Việt hiện còn hạn chế về quy mô, chất lượng và tính đa dạng, đặc biệt là dữ liệu viết tay. Chưa có nhiều công trình phân tích khả năng thích ứng của VLM khi được tinh chỉnh bằng kỹ thuật LoRA cũng như chưa có đánh giá so sánh trực tiếp giữa mô hình VLM và mô hình OCR truyền thống trên cùng tập dữ liệu tiếng Việt. Đây là một lý do lớn tạo ra khoảng trống nghiên cứu đáng kể và là động lực thúc đẩy thực hiện đề án.

Vì vậy, đề án **"Ứng dụng mô hình thị giác–ngôn ngữ lớn vào bài toán nhận dạng và trích xuất thông tin từ ảnh tài liệu tiếng Việt"** được thực hiện nhằm nghiên cứu và đánh giá khả năng OCR của mô hình Qwen 2.5-VL đối với tiếng Việt. Đề án triển khai các bước từ chuẩn bị dữ liệu và tiền xử lý dữ liệu (bao gồm chữ in khó và chữ viết tay), chuẩn hóa định dạng theo phong cách hội thoại phù hợp với VLM, xây dựng pipeline tinh chỉnh bằng LoRA, cho đến triển khai đánh giá mô hình trên bộ benchmark được thiết kế đầy đủ. Ngoài ra, để đảm bảo tính khách quan, mô hình cũng được so sánh với PaddleOCR – một trong những hệ thống OCR

truyền thống mạnh nhất hiện nay – và Vintern – một mô hình VLM nhẹ hơn có khả năng đọc tài liệu.

Nội dung của đề án được tổ chức thành ba chương chính như sau:

- **Chương 1** giới thiệu tổng quan về bài toán nhận dạng ký tự tiếng Việt, phân tích những hạn chế còn tồn tại trong các phương pháp OCR truyền thống và tổng hợp các hướng tiếp cận đã được nghiên cứu, đặc biệt là các mô hình thị giác–ngôn ngữ hiện đại (VLM).
- **Chương 2** trình bày phương pháp đề xuất, bao gồm quy trình xây dựng và chuẩn hóa dữ liệu huấn luyện, thiết kế định dạng dữ liệu phù hợp với mô hình, cũng như mô tả chi tiết pipeline tinh chỉnh Qwen 2.5-VL bằng kỹ thuật LoRA và bộ tiêu chí đánh giá được sử dụng.
- **Chương 3** mô tả quá trình triển khai thực nghiệm, từ việc chuẩn bị dữ liệu, lựa chọn các mô hình đối sánh (baseline), đến đánh giá theo các độ đo CER, WER và tốc độ xử lý. Chương này đồng thời phân tích hiệu quả của mô hình sau khi tinh chỉnh và trình bày ứng dụng demo minh họa khả năng nhận dạng và trích xuất thông tin từ ảnh tài liệu.

CHƯƠNG 1: TỔNG QUAN VỀ BÀI TOÁN

1.1. Giới thiệu chung về bài toán OCR

Nhận dạng ký tự quang học (Optical Character Recognition – OCR) là bài toán chuyển đổi nội dung văn bản xuất hiện trong hình ảnh (scan tài liệu, ảnh chụp, biển hiệu, màn hình, v.v.) thành chuỗi ký tự có thể xử lý tự động. Với sự phát triển mạnh mẽ của các hệ thống quản lý tài liệu, chính phủ điện tử, ngân hàng số và thương mại điện tử, OCR trở thành một thành phần hạ tầng quan trọng trong nhiều hệ thống thực tế như: số hóa hồ sơ giấy, tự động nhập liệu biểu mẫu, trích xuất thông tin hóa đơn, tìm kiếm tài liệu theo nội dung, kiểm soát tuân thủ và lưu trữ lâu dài.

Đối với tiếng Anh và các ngôn ngữ Latinh không dấu, OCR đã đạt độ chính xác rất cao trong nhiều kịch bản, thậm chí được coi là được giải quyết hoàn toàn trong những điều kiện chuẩn hóa (scan sắc nét, font chữ tiêu chuẩn, bố cục rõ ràng) [2]. Tuy nhiên, khi mở rộng sang các ngôn ngữ có hệ thống dấu phức tạp hoặc dạng chữ viết tay tự nhiên, bài toán vẫn còn nhiều thách thức [3].

Với tiếng Việt, OCR có vai trò đặc biệt quan trọng do lượng tài liệu hành chính, biểu mẫu, sổ sách và hóa đơn khổng lồ đang tồn tại dưới dạng giấy. Nhiều cơ quan nhà nước và doanh nghiệp đang triển khai chuyển đổi số, trong đó số hóa tài liệu giấy là bước đầu tiên để có thể áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) và khai phá dữ liệu ở các bước sau. Các khảo sát gần đây về phân tích và nhận dạng tài liệu tiếng Việt (Vietnamese Document Analysis and Recognition – DAR) [1] cho thấy OCR vẫn là một trong những "mắt xích" hạn chế hiệu quả tổng thể của hệ thống, đặc biệt với tài liệu thực tế chất lượng kém hoặc chữ viết tay.

Trong bối cảnh đó, việc nghiên cứu, đánh giá và cải tiến các phương pháp OCR cho tiếng Việt, đặc biệt trên dữ liệu thực tế (ảnh chụp, tài liệu scan lệch, chữ viết tay, font lạ), là một hướng nghiên cứu có ý nghĩa cả về lý luận và ứng dụng.

1.2. Đặc thù tiếng Việt trong bài toán OCR

1.2.1. Hệ thống chữ viết và dấu tiếng Việt

Tiếng Việt sử dụng bảng chữ cái Latinh mở rộng với 29 chữ cái cơ bản, kết hợp với 5 dấu thanh (sắc, huyền, hỏi, ngã, nặng) và nhiều dấu phụ trên – dưới (â, ê, ô, ă, ơ, ư, đ). Khi kết hợp cả dấu thanh và dấu phụ, số "hình dạng ký tự" có thể phân biệt trong ảnh tăng lên rất lớn. Một số nghiên cứu về nhận dạng ký tự tiếng Việt cho thấy số lớp ký tự có thể lên tới hơn 150–170 nếu phân biệt từng biến thể ký tự–dấu riêng biệt [1].

Điều này khiến mô hình OCR khó học hơn so với các ngôn ngữ Latinh đơn giản, bởi khoảng cách thị giác giữa hai ký tự khác nhau đôi khi rất nhỏ (ví dụ: "a" – "à" – "ả" – "ã" – "ạ", hoặc "o" – "ô" – "ơ"). Khi ảnh mờ, nhiễu, hoặc độ phân giải thấp, các dấu dễ bị mờ hoặc "dính" vào nền, dẫn tới lỗi nhận dạng.

1.2.2. Bố cục tài liệu và đa dạng nguồn dữ liệu

Tài liệu tiếng Việt trong thực tế đến từ nhiều nguồn khác nhau:

- Tài liệu hành chính (giấy khai sinh, sổ hộ khẩu, CCCD cũ, đơn từ, biên bản)
- Biểu mẫu in (phiếu đăng ký, biểu mẫu thuế, mẫu khai báo)
- Hóa đơn, phiếu thu, phiếu chi, phiếu xuất–nhập kho
- Sách giáo khoa, tài liệu học tập, luận văn, báo cáo
- Văn bản chữ viết tay (phiếu đăng ký, đơn viết tay, bài thi, phiếu chấm điểm)

Các nguồn này có sự khác biệt lớn về:

- Font chữ: từ font hành chính, font máy in cũ, font không Unicode đến font mới.
- Bố cục: có tài liệu chuẩn hóa, có tài liệu bố cục tự do hoặc thay đổi theo từng cơ quan.

- Chất lượng hình ảnh: scan trắng đen, scan lệch góc, ảnh chụp bằng điện thoại trong điều kiện ánh sáng không đều, có bóng, nhiễu, mờ.

Những yếu tố này làm cho bài toán OCR tiếng Việt không chỉ là nhận dạng ký tự, mà còn là bài toán thích ứng với dữ liệu thực tế đa dạng và phức tạp.

1.2.3. Chữ in và chữ viết tay

Đối với chữ in, cấu trúc ký tự tương đối ổn định, mô hình OCR có thể đạt độ chính xác cao khi dữ liệu đủ sạch. Tuy nhiên, chữ viết tay tiếng Việt có độ biến thiên hình dạng rất lớn giữa các cá nhân, khác biệt về:

- Độ nghiêng, độ đậm–nhạt, nét nối giữa các ký tự
- Thói quen viết tắt, lược bỏ dấu hoặc ghi dấu không chuẩn vị trí
- Độ nghiêng của dòng chữ, khoảng cách giữa các từ không đều

Do đó, OCR chữ viết tay thường được xem là một bài toán khó hơn đáng kể so với OCR chữ in. Các cuộc thi như Kalapa Challenge [4] về nhận dạng chữ viết tay tiếng Việt, hay các bộ dữ liệu VNOnDB, UIT-HWDB [5]... đã được xây dựng nhằm thúc đẩy nghiên cứu trong lĩnh vực này.

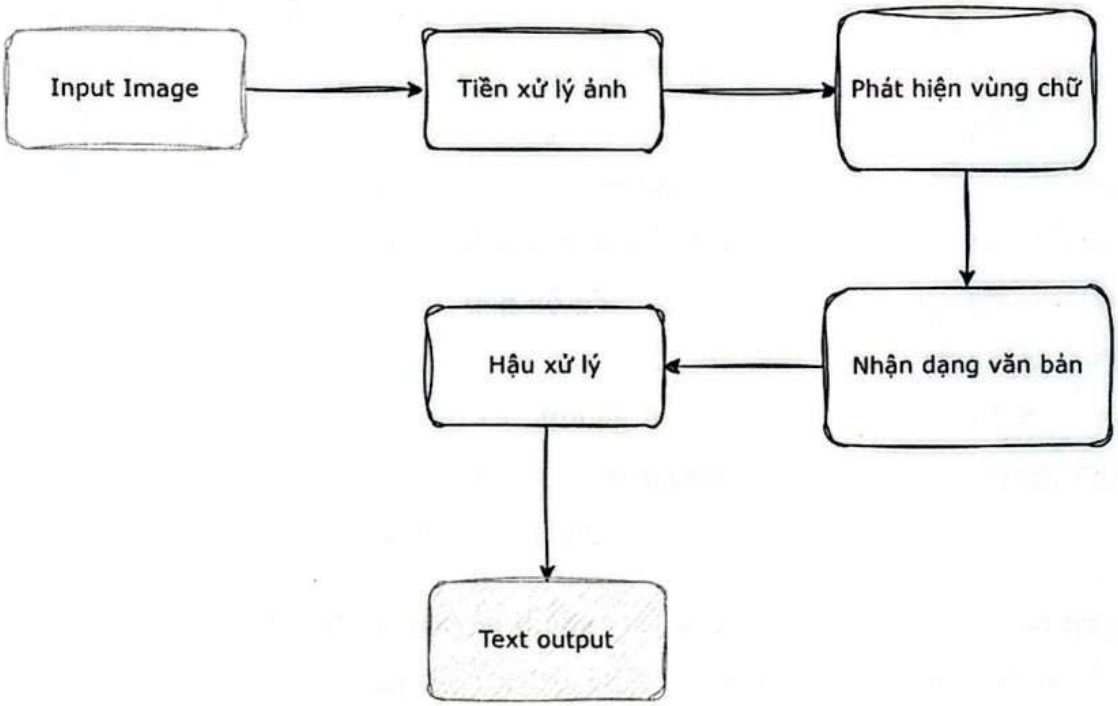
1.3. Các hướng tiếp cận OCR hiện nay

1.3.1. Pipeline OCR truyền thống

Các hệ thống OCR kinh điển như Tesseract [6], PaddleOCR [8] hay các giải pháp thương mại thường tuân theo pipeline ba bước chính:

1. **Tiền xử lý ảnh**: chuyển ảnh sang grayscale, nhị phân hóa, hiệu chỉnh độ nghiêng, loại bỏ nhiễu, chuẩn hóa kích thước.
2. **Phát hiện vùng chữ (text detection)**: xác định vị trí các khối văn bản, dòng, từ hoặc ký tự trong ảnh.

3. **Nhận dạng văn bản (text recognition)**: biến vùng ảnh chứa chữ thành chuỗi ký tự.
4. **Hậu xử lý**: dùng từ điển, mô hình ngôn ngữ hoặc heuristic để sửa lỗi, chuẩn hóa kết quả.



Hình 1.1: Pipeline OCR truyền thống

Tesseract (phiên bản 4 trở đi) sử dụng mạng LSTM để nhận dạng chuỗi ký tự từ ảnh văn bản đã được chuẩn hóa. Cụ thể, ảnh đầu vào sau khi trải qua tiền xử lý (chuyển sang thang xám, chuẩn hóa kích thước, hiệu chỉnh độ nghiêng…) sẽ được biến đổi thành chuỗi feature thông qua một mạng CNN nông. Chuỗi đặc trưng này được đưa vào mô hình LSTM hai chiều (BiLSTM) nhằm mô hình hóa quan hệ phụ thuộc theo trục ngang của văn bản. Cuối cùng, thuật toán CTC (Connectionist Temporal Classification) được áp dụng để ánh xạ chuỗi biểu diễn có độ dài thay đổi thành chuỗi ký tự đầu ra, cho phép mô hình học được sự liên tục của ký tự mà không cần phải gán nhãn cắt (segmentation) theo từng ký tự riêng lẻ. Nhờ cơ chế LSTM +

CTC, Tesseract 4 hoạt động hiệu quả hơn đáng kể so với các phiên bản trước vốn dựa trên phân đoạn ký tự truyền thống.

Trong khi đó, PaddleOCR cung cấp một bộ pipeline hiện đại hơn, được thiết kế theo kiến trúc modular với nhiều lựa chọn mô hình. Ở bước phát hiện văn bản (text detection), PaddleOCR sử dụng các backbone mạnh như ResNet kèm theo các detector như DB (Differentiable Binarization) hoặc EAST. DB đặc biệt hiệu quả trong nhận dạng văn bản ở môi trường thực tế nhờ cơ chế phân ngưỡng mềm (adaptive binarization) giúp mô hình phát hiện chính xác các vùng chữ có biên yếu hoặc nền phức tạp. Ở bước nhận dạng văn bản (text recognition), PaddleOCR cung cấp nhiều mô hình phù hợp cho từng độ phức tạp của bài toán: kiến trúc CRNN (CNN + RNN + CTC) cho các trường hợp văn bản thẳng, rõ ràng; hoặc Transformer-based recognizer cho các trường hợp văn bản khó, nhiều biến dạng hoặc có ký tự nhỏ. Các mô hình này đều hỗ trợ đa ngôn ngữ, trong đó có tiếng Việt, và có thể fine-tune trên tập dữ liệu tùy chỉnh.

Ưu điểm của pipeline OCR truyền thống xuất phát từ việc các thành phần được thiết kế tách biệt và có thể thay thế độc lập. Điều này mang lại ba lợi thế chính:

- **Dễ triển khai**, do nhiều thư viện, mô hình tham chiếu và tài liệu hướng dẫn đã được xây dựng trong thời gian dài (Tesseract, OCRopus, PaddleOCR…).
- **Dễ tùy biến**, vì có thể thay thế detector (EAST → DB) hoặc recognizer (CRNN → Transformer) mà không ảnh hưởng đến các phần còn lại của pipeline.
- **Hiệu quả tốt trên tài liệu tiêu chuẩn**, như scan sách giáo khoa, báo chí, tài liệu văn phòng rõ ràng, nơi bố cục ít phức tạp và chất lượng hình ảnh ổn định.

Tuy nhiên, khi áp dụng cho tiếng Việt – một ngôn ngữ có hệ thống dấu phức tạp – pipeline truyền thống bộc lộ nhiều hạn chế:

- **Sự phụ thuộc mạnh vào phát hiện văn bản (text detection)**: nếu detector cắt thiếu dấu (dấu sắc/huyền/nặng) hoặc cắt chồng lên ký tự bên cạnh, recognizer gần như không thể nhận diện chính xác do tiếng Việt nhạy với mất dấu.
- **Văn bản cong, xiên hoặc chụp trong môi trường thực tế** khiến bộ phát hiện khó có thể xác đinh đúng bounding box, kéo theo kết quả nhận dạng bị sai hàng loạt.
- **Kích thước dấu tiếng Việt nhỏ** khiến dấu dễ bị mất trong các bước nhị phân hóa, resize, hoặc sharpening — đây là vấn đề đặc biệt nghiêm trọng trên ảnh chụp từ điện thoại hoặc tài liệu scan chất lượng thấp.
- **Bố cục phức tạp**, như tài liệu đa cột, bảng biểu, form điện tử, văn bản kèm watermark, stamp… khiến detector phát hiện sai vùng văn bản hoặc gộp nhầm các dòng không liên quan, làm giảm độ chính xác tổng thể.

1.3.2. OCR dựa trên học sâu

Các nghiên cứu gần đây trên thế giới cho thấy xu hướng chuyển dần từ pipeline tách biệt sang mô hình end-to-end dựa trên deep learning, trong đó:

- Phần trích xuất đặc trưng dùng CNN sâu (ResNet, VGG, EfficientNet…).
- Phần mô hình hóa chuỗi dùng RNN/LSTM, GRU, hoặc Transformer.
- Loss sử dụng CTC hoặc attention-based seq2seq.

Các mô hình như CRNN [10], SATRN [11], Rosetta [12], TrBA [13] đã được áp dụng rộng rãi cho scene nhận dạng văn bản (text recognition).

Trong cộng đồng tiếng Việt, một ví dụ tiêu biểu là thư viện **VietOCR** [14], áp dụng kiến trúc Transformer OCR (CNN + Transformer Encoder–Decoder) để nhận dạng tiếng Việt với khả năng huấn luyện lại (fine-tune) cho từng domain.

1.4. Các nghiên cứu về OCR tiếng Việt

1.4.1. Nhận dạng chữ in và văn bản in trong tài liệu

Nhiều nghiên cứu tập trung vào OCR cho chữ in tiếng Việt, đặc biệt trong bối cảnh sách giáo khoa, tài liệu in và tài liệu scan:

- Một số công trình xây dựng mô hình CNN để nhận dạng ký tự tiếng Việt với số lớp ký tự giảm bớt (gom nhóm các biến thể dấu) nhằm giảm độ phức tạp, sau đó tái gán dấu ở bước hậu xử lý. Các nghiên cứu này thường thử nghiệm trên dữ liệu scan từ sách giáo khoa hoặc tài liệu in, cho thấy mô hình CNN có thể nhận dạng tốt ngay cả khi số lớp ký tự lớn (trên 100 lớp) [2].
- Các hệ thống thực tế như VietOCR (GUI dựa trên Tesseract), các dự án mã nguồn mở kết hợp PaddleOCR và VietOCR trên GitHub cho thấy cộng đồng đã quan tâm mạnh tới việc xây dựng pipeline OCR tối ưu cho tiếng Việt, trong đó có thể thay thế recognizer mặc định bằng mô hình được huấn luyện riêng cho tiếng Việt [14].

Các bộ dữ liệu công khai cho OCR tài liệu tiếng Việt còn hạn chế, tuy nhiên một số dataset liên quan như UIT-ViODC, ViTextVQA, Viet-OCR-VQA đã được công bố, cung cấp ảnh tài liệu có chứa văn bản tiếng Việt kèm nhãn nội dung, tạo nền tảng cho việc huấn luyện và đánh giá các mô hình OCR/VQA cho tiếng Việt [5][15].

1.4.2. Nhận dạng chữ trong ảnh cảnh

Bên cạnh tài liệu scan, một nhánh nghiên cứu khác tập trung vào nhận dạng chữ trong ảnh cảnh (biển hiệu, poster, biển quảng cáo, video). Nhiều công trình đã:

- Thu thập và công bố các bộ dữ liệu như VinText, VnSceneText, chứa các ảnh chụp thực tế trong môi trường đô thị, với chữ tiếng Việt ở nhiều góc nhìn, ánh sáng và font chữ khác nhau [16].
- Áp dụng các mô hình detection hiện đại như EAST, SAST, DB kết hợp với SATRN, Rosetta hoặc CRNN để nhận dạng chuỗi, từ đó đánh giá hiệu quả của các kiến trúc khác nhau trên dữ liệu tiếng Việt.

- Đề xuất các khung ensemble hoặc framework kết hợp nhiều mô hình để tăng độ chính xác trong bối cảnh văn bản nhỏ, mờ, bị che khuất một phần, hoặc xuất hiện trong video streaming.
- Các nghiên cứu về văn bản trong cảnh chụp thực tế cho tiếng Việt thường cho kết quả tốt hơn trên các biển hiệu rõ ràng, nhưng vẫn gặp khó khăn khi kích thước chữ nhỏ, độ tương phản thấp, hoặc chữ cách điệu mạnh.

1.4.3. Nhận dạng chữ viết tay tiếng Việt

Nhận dạng chữ viết tay tiếng Việt là một bài toán khó hơn chữ in do tính đa dạng về nét chữ, cách đặt dấu, và thói quen viết tắt. Một số hướng tiếp cận và dữ liệu tiêu biểu:

- VNOnDB-Line: bộ dữ liệu chữ viết tay online tiếng Việt, trong đó tín hiệu bút (tọa độ, thời gian) được ghi lại, phục vụ cho các mô hình sử dụng BLSTM và các kiến trúc RNN khác để nhận dạng chữ viết tay online.
- Cinnamon AI / Kalapa Handwritten Challenge: cuộc thi nhận dạng dòng chữ viết tay tiếng Việt (địa chỉ, thông tin cá nhân) với dữ liệu ảnh offline (scan ảnh hoặc ảnh chụp), khuyến khích sử dụng các mô hình CRNN, CNN–LSTM, kết hợp CTC. Nhiều nhóm đã công bố kết quả thực nghiệm cho thấy mô hình CRNN với attention hoặc Transformer-based recognizer cho kết quả khả quan trên dữ liệu này.
- Một số nghiên cứu ứng dụng OCR chữ viết tay vào các bài toán cụ thể, chẳng hạn trích xuất thông tin từ phiếu đăng ký tuyển sinh trung học phổ thông, biểu mẫu đăng ký, đơn viết tay…, thường sử dụng pipeline: phát hiện vùng chứa dòng chữ → nhận dạng dòng chữ bằng CRNN/CTC → hậu xử lý bằng các quy tắc hoặc mô hình ngôn ngữ.

Ngoài ra, cộng đồng cũng xây dựng nhiều tập dữ liệu nhỏ hơn về chữ viết tay trên Kaggle và GitHub nhằm hỗ trợ thực nghiệm nhanh, ví dụ như bộ dữ liệu

Vietnamese Handwritten OCR chứa các từ hoặc ký tự riêng lẻ, phục vụ nhận dạng ở mức word/character.

Nhìn chung, OCR chữ viết tay tiếng Việt đang trong giai đoạn phát triển, chưa đạt độ chín như chữ in. Các thách thức chính gồm: thiếu dữ liệu lớn và đa dạng, sự khác biệt giữa các kiểu chữ, và tính "khó chuẩn hóa" của cách ghi dấu trong thực tế (đặt dấu lộn vị trí, viết thiếu dấu, v.v.).

1.5. Xu hướng mô hình thị giác–ngôn ngữ và OCR tài liệu tiếng Việt

Trong những năm gần đây, các mô hình thị giác–ngôn ngữ lớn như LayoutLM, Donut, Pix2Struct, GIT, Qwen-VL… đã nổi lên như một hướng tiếp cận đầy tiềm năng cho bài toán hiểu tài liệu và nhận dạng văn bản. Các mô hình này ban đầu được phát triển cho nhiều nhiệm vụ đa dạng, từ sinh mô tả ảnh (image captioning), trả lời câu hỏi dựa trên ảnh (VQA), phân tích tài liệu (document understanding) cho đến xử lý giao diện người dùng (UI screenshot). Tuy nhiên, chính khả năng kết hợp trực tiếp giữa thị giác và ngôn ngữ đã giúp chúng trở thành lựa chọn mới cho bài toán OCR tài liệu, đặc biệt trong những tình huống mà các pipeline detection–recognition truyền thống gặp hạn chế.

Một đặc điểm nổi bật của VLM là khả năng hiểu ngữ cảnh toàn cục (global context) của trang tài liệu. Thay vì xử lý từng ký tự, từ hoặc dòng một cách độc lập, mô hình có thể phân tích mối quan hệ giữa các khối văn bản, tiêu đề, chú thích, hình minh họa, bảng biểu và các thành phần bố cục khác. Điều này cho phép mô hình tránh được những lỗi thường gặp trong OCR truyền thống, chẳng hạn như nhận dạng sai do bounding box bị cắt lệch, hoặc nhầm lẫn thứ tự đọc khi tài liệu có bố cục nhiều cột.

Bên cạnh đó, nhiều mô hình VLM hiện đại đã hỗ trợ trích xuất văn bản trực tiếp từ ảnh mà không cần module detection, dựa trên cơ chế sinh văn bản (generative decoding) tương tự như LLM. Ví dụ, các kiến trúc như Donut có thể "đọc" cả trang

tài liệu theo hướng tuần tự và sinh ra nội dung dưới dạng text hoặc JSON. Hướng tiếp cận "OCR-free" này giảm thiểu lỗi tích lũy qua nhiều bước và phù hợp với các tài liệu phức tạp, bị mờ, thiếu hụt cấu trúc hoặc được chụp trong điều kiện thực tế không chuẩn hóa.

Một số mô hình VLM còn hỗ trợ Document-VQA (question answering trên tài liệu), cho phép người dùng truy xuất thông tin mà không cần xác định vị trí của vùng chữ. Điều này đặc biệt hữu ích trong các ứng dụng tự động hóa quy trình (RPA), khi hệ thống cần trả lời các câu hỏi như "Số hóa đơn là gì?", "Ngày ký trên văn bản?", "Tên khách hàng nằm ở đâu?" mà không phải xử lý OCR toàn bộ tài liệu.

Survey "A Survey on Vietnamese Document Analysis and Recognition: Challenges and Future Directions" [2] đã cung cấp cái nhìn tổng quan về thực trạng phân tích tài liệu tiếng Việt. Trong đó, tác giả nhấn mạnh rằng OCR là một trong những điểm nghẽn quan trọng nhất khi xây dựng hệ thống DAR hoàn chỉnh. Survey cũng chỉ ra rằng sự xuất hiện của LLM và VLM mở ra cơ hội đáng kể để giải quyết vấn đề thiếu dữ liệu tiếng Việt, cải thiện độ chính xác trong môi trường thực tế nhiều nhiễu và nâng cao khả năng hiểu ngữ cảnh của mô hình.

Trong số các mô hình hiện đại, Qwen 2.5-VL [17] là đại diện tiêu biểu cho thế hệ VLM mới, được huấn luyện trên dữ liệu đa phương thức quy mô lớn và hỗ trợ tốt tiếng Việt nhờ kho dữ liệu multilingual phong phú. Mô hình có một số đặc điểm nổi bật:

- Xử lý ảnh tài liệu độ phân giải cao, hỗ trợ tài liệu có nhiều cột, nhiều vùng nội dung khác nhau mà vẫn giữ được chi tiết.
- Khả năng hiểu ngữ cảnh toàn trang, giúp mô hình phân biệt chính xác mối quan hệ giữa các thành phần trong tài liệu và sắp xếp trình tự đọc hợp lý.

- Hỗ trợ tốt tiếng Việt, bao gồm nhận dạng từ có dấu, ký tự đặc biệt và các trường hợp dễ gây nhầm lẫn trong OCR truyền thống.
- Cho phép tinh chỉnh (SFT/LoRA) dễ dàng, thuận lợi khi áp dụng cho tập dữ liệu OCR chuyên biệt theo domain như tài liệu hành chính, hóa đơn, hồ sơ ngân hàng, biểu mẫu nhập liệu,…
- Tương thích với pipeline trích xuất thông tin, hỗ trợ sinh text, JSON hoặc trả lời câu hỏi trên tài liệu dưới dạng tự nhiên (natural-language QA).

Những ưu điểm trên khiến VLM trở thành xu hướng triển khai OCR thế hệ mới, đặc biệt trong bối cảnh tiếng Việt thiếu dữ liệu lớn, bố cục tài liệu thường phức tạp và có nhiều biến thể về chất lượng hình ảnh.

Đây cũng chính là định hướng của đề tài: ứng dụng và tinh chỉnh mô hình thị giác–ngôn ngữ lớn Qwen 2.5-VL 7B cho bài toán nhận dạng và trích xuất nội dung từ ảnh tài liệu tiếng Việt. Sau đó, mô hình được so sánh với hai nhóm baseline tiêu biểu:

- OCR truyền thống như PaddleOCR, nhằm đánh giá mức độ cải thiện khi chuyển từ pipeline detection–recognition sang mô hình end-to-end đa phương thức.
- Mô hình VLM khác (Vintern) để phân tích ưu – nhược điểm tương đối giữa các mô hình VLM khi áp dụng cho OCR tiếng Việt.

Kết quả kỳ vọng là làm rõ lợi thế của mô hình thị giác–ngôn ngữ trong việc xử lý ngữ cảnh, giảm lỗi và cải thiện hiệu suất OCR trên tài liệu tiếng Việt thực tế.

1.6. Đánh giá chung và khoảng trống nghiên cứu

Từ tổng quan trên có thể rút ra một số nhận định:

- OCR tiếng Việt đã có nhiều tiến bộ, cả ở mức chữ in (tài liệu, sách, hóa đơn) lẫn văn bản trong cảnh chụp thực tế (biển hiệu) và chữ viết tay. Các thư viện

và mô hình như Tesseract, PaddleOCR, VietOCR, CRNN/Transformer đã được áp dụng khá rộng rãi trong cộng đồng.

- Tuy nhiên, độ chính xác trên dữ liệu thực tế phức tạp vẫn chưa ổn định, đặc biệt khi:
    - Ảnh mờ, nhiễu, mất nét, bị lệch góc.
    - Dấu tiếng Việt nhỏ, dễ bị mất hoặc dính.
    - Bố cục tài liệu không chuẩn, nhiều bảng biểu, nhiều cột.
    - Chữ viết tay với nét chữ rất đa dạng.
- Dữ liệu công khai cho OCR tiếng Việt còn rời rạc, quy mô vừa và nhỏ, ít có bộ dữ liệu chuẩn hóa cho bài toán tài liệu hành chính hoặc biểu mẫu thực tế. Một số dataset mới như Viet-OCR-VQA, ViTextVQA mở ra hướng tiếp cận VQA trên tài liệu nhưng vẫn chưa được khai thác triệt để cho OCR thuần túy.
- Xu hướng sử dụng VLM và LLM cho OCR tiếng Việt mới ở giai đoạn đầu. Các survey gần đây chủ yếu mang tính tổng hợp và đề xuất hướng nghiên cứu; chưa có nhiều công bố thực nghiệm chi tiết về việc tinh chỉnh một VLM cỡ lớn như Qwen 2.5-VL 7B trên dataset OCR tiếng Việt và so sánh có hệ thống với các baseline truyền thống.

Khoảng trống này là cơ hội để đề tài thực hiện:

- Xây dựng và chuẩn hóa một tập dữ liệu OCR tiếng Việt "khó" (chữ in, chữ viết tay, font lạ, ảnh chất lượng kém).
- Thiết kế pipeline tinh chỉnh (SFT/LoRA) cho Qwen 2.5-VL 7B trên dữ liệu này.
- So sánh định lượng với PaddleOCR và một VLM khác (Vintern 1B) theo các chỉ số CER, WER, thời gian suy luận, khả năng tổng quát hóa.
- Phân tích ưu – nhược điểm của hướng VLM-based OCR cho tiếng Việt, từ đó đề xuất hướng phát triển trong tương lai.

1.7. Kết luận chương

Chương 1 đã trình bày tổng quan về bài toán nhận dạng ký tự quang học (OCR) cho tài liệu tiếng Việt, từ đặc thù ngôn ngữ, đặc điểm dữ liệu thực tế đến các hướng tiếp cận chính hiện nay. Các nghiên cứu về OCR chữ in, văn bản trong cảnh chụp thực tế và chữ viết tay tiếng Việt cho thấy deep learning – đặc biệt là các kiến trúc CNN–RNN, Transformer – đã giúp nâng cao đáng kể chất lượng nhận dạng. Tuy vậy, những hạn chế về dữ liệu, độ phức tạp của dấu tiếng Việt và sự đa dạng của tài liệu thực tế vẫn là rào cản lớn.

Bên cạnh đó, xu hướng mới sử dụng các mô hình thị giác–ngôn ngữ lớn (VLM) mở ra một hướng tiếp cận đầy tiềm năng, cho phép mô hình xử lý tài liệu một cách toàn diện hơn, hiểu ngữ cảnh toàn trang, và có khả năng thích ứng tốt hơn với dữ liệu thực tế. Việc tinh chỉnh các mô hình như Qwen 2.5-VL 7B cho OCR tiếng Việt, và đánh giá có hệ thống so với các baseline truyền thống, là một đóng góp mới cả về mặt thực nghiệm lẫn ứng dụng, đồng thời tạo nền tảng cho các chương tiếp theo của đề án.

CHƯƠNG 2: PHƯƠNG PHÁP ĐỀ XUẤT

2.1. Định hướng và mục tiêu của phương pháp

Dựa trên tổng quan ở Chương 1, có thể thấy các phương pháp OCR truyền thống (Tesseract, PaddleOCR, CRNN, Transformer OCR...) tuy đã đạt hiệu quả tốt trên nhiều loại dữ liệu, nhưng vẫn gặp khó khăn đáng kể với tài liệu tiếng Việt phức tạp, đặc biệt là khi chất lượng ảnh thấp, bố cục không chuẩn hoặc có chữ viết tay. Đồng thời, sự xuất hiện của các mô hình thị giác–ngôn ngữ lớn (Vision–Language Models – VLMs) như Qwen 2.5-VL cho phép xử lý trực tiếp ảnh tài liệu theo hướng sinh văn bản (image-to-text generation), bỏ qua pipeline tách biệt với nhiều bước từ detection tới recognition.

Trên cơ sở đó, phương pháp đề xuất trong đề án này theo định hướng:

- Tận dụng mô hình VLM đó là Qwen 2.5-VL 7B như một bộ mã hóa – sinh văn bản đa phương thức, nhận trực tiếp ảnh tài liệu tiếng Việt và sinh ra nội dung văn bản tương ứng.
- Tinh chỉnh (fine-tuning) mô hình bằng kỹ thuật LoRA trên bộ dữ liệu OCR tiếng Việt đã chuẩn hóa (bao gồm chữ in khó và chữ viết tay), nhằm giúp mô hình thích ứng tốt hơn với đặc thù tiếng Việt và dữ liệu thực tế.
- So sánh có hệ thống với các baseline mạnh hiện nay gồm PaddleOCR (pipeline OCR truyền thống), Vintern (một VLM khác), và Qwen 2.5-VL trước khi tinh chỉnh.

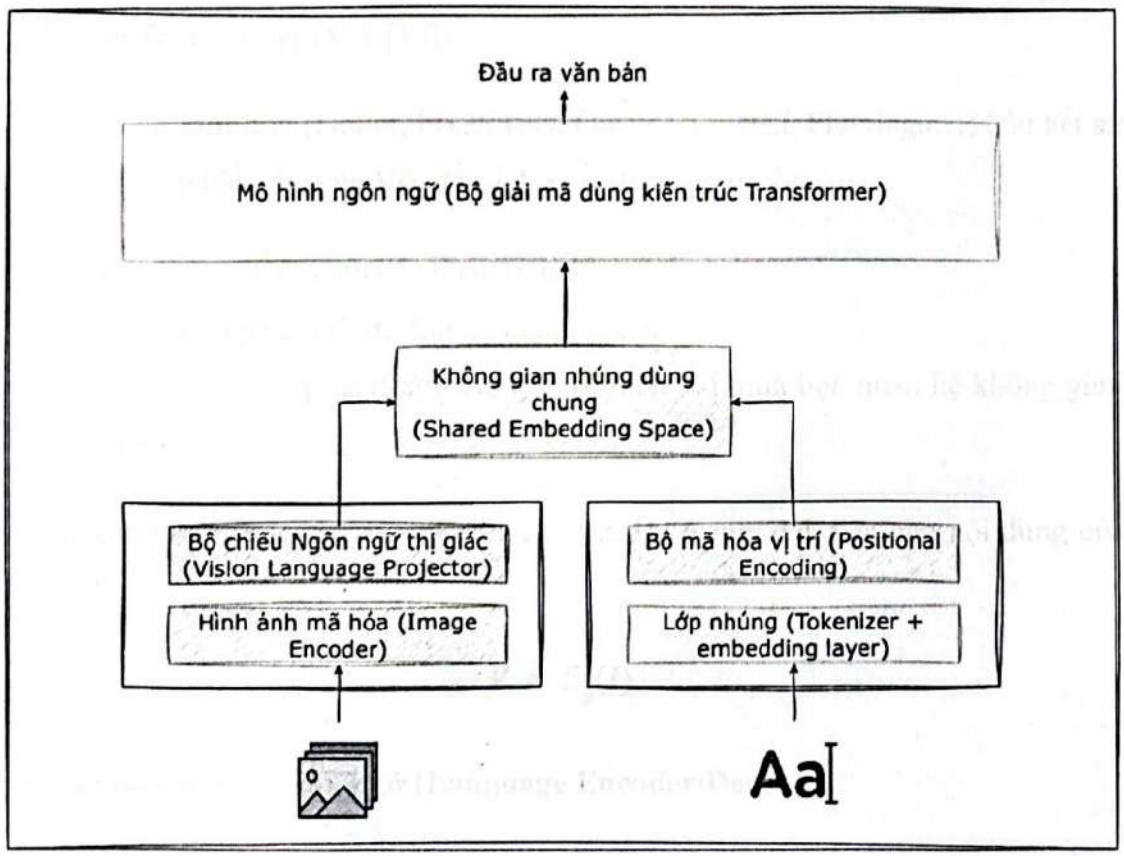
Mục tiêu cuối cùng là xây dựng một **pipeline OCR dựa trên VLM** có khả năng:

- Nhận dạng văn bản tiếng Việt với độ chính xác được cải thiện (đo qua các độ đo CER, WER).
- Thấy được ưu thế hơn so với pipeline truyền thống trên các tập dữ liệu "khó" (mờ, nhiễu, chữ viết tay, font lạ).

2.2. Cơ sở lý thuyết

2.2.1. Cơ sở lý thuyết về mô hình thị giác–ngôn ngữ (VLM)

Mô hình thị giác–ngôn ngữ (Vision–Language Models – VLMs) là một nhánh quan trọng trong học sâu đa mô thức, được thiết kế để xử lý đồng thời dữ liệu hình ảnh và văn bản – hai nguồn thông tin có bản chất và cấu trúc hoàn toàn khác nhau. Mục tiêu cốt lõi của VLM là xây dựng không gian biểu diễn chung (shared representation space), trong đó đặc trưng thị giác và đặc trưng ngôn ngữ có thể tương tác, bổ trợ lẫn nhau thông qua các cơ chế học đa mô thức.



Hình 2.1: Kiến trúc chung của mô hình VLM

Về mặt kiến trúc, phần lớn các VLM hiện đại có thể được khái quát thành ba thành phần chính.

Bộ mã hóa hình ảnh (Visual Encoder)

Bộ mã hóa hình ảnh (visual encoder) chịu trách nhiệm trích xuất thông tin không gian – hình dạng – bố cục từ ảnh đầu vào. Hai hướng tiếp cận chính:

(a) CNN-based encoder

Các mô hình cổ điển như ResNet, EfficientNet và ConvNeXt thường được sử dụng khi cần kiến trúc nhẹ, có tốc độ suy luận nhanh. CNN phù hợp với bài toán phát hiện ngữ cảnh văn bản (scene text), OCR tài liệu nhẹ và các tác vụ cần thông tin cục bộ mạnh.

(b) Vision Transformer (ViT [19])

Các mô hình mới hơn (Donut, Pix2Struct, Qwen-VL, PaLI, Flamingo…) hầu hết sử dụng ViT hoặc biến thể của ViT để trích xuất đặc trưng:

- Ảnh được chia thành các phần (patch)
- Mỗi phần (patch) được ánh xạ thành token
- Cơ chế tự chú ý (self-attention) cho phép mô hình học quan hệ không gian dài hạn

Kết quả là visual encoder tạo ra chuỗi token đặc trưng, đại diện cho nội dung của hình ảnh

$$V = E_v(I)$$

Bộ mã hóa/giải mã ngôn ngữ (Language Encoder/Decoder)

Nhánh xử lý văn bản trong VLM thường dựa trên Backbone Transformer, bao gồm:

- Encoder-only như BERT, RoBERTa (LayoutLM)
- Decoder-only (LLM) như GPT, LLaMA, Qwen (Qwen-VL, Kosmos)
- Encoder–decoder như T5, mT5 (Pix2Struct, PaLI)

Chức năng chính:

- Mã hóa văn bản đầu vào thành vector biểu diễn
- Học ngữ cảnh, cú pháp, nghĩa
- Sinh văn bản đầu ra (đối với decoder-based hoặc seq2seq)

Nếu ta ký hiệu văn bản đầu vào là T, thì:

$$L = E_l(T) \text{ là biểu diễn của văn bản tương ứng}$$

Cơ chế kết hợp đa mô thức (Multimodal Fusion / Alignment)

Đây là thành phần quan trọng nhất của VLM [20], quyết định mô hình có thực sự hiểu và liên kết được thông tin giữa hai mô thức hay không. Ba dạng kết hợp phổ biến:

(a) Projection-based fusion

Dùng một lớp tuyến tính để chiếu visual tokens vào không gian embedding của language model.

→ Đơn giản, nhẹ, phù hợp để gắn visual encoder vào LLM (như Qwen-VL [22], LLaVA [21]).

(b) Cross-attention

Tokens từ văn bản và hình ảnh tham gia vào cơ chế attention hai chiều, nơi:

- Text tokens có thể "nhìn" visual tokens
- Visual tokens có thể "nhìn" lại text tokens

Cách này tạo sự hội tụ sâu giữa hai mô thức và phù hợp các tác vụ cần hiểu bố cục (layout-aware).

(c) Late fusion / Early fusion

- Early fusion: trộn đặc trưng ngay từ đầu (thường dùng trong các mô hình cực lớn).
- Late fusion: nối hoặc ghép kết quả của hai encoder rồi mới xử lý chung.

Tổng quát, quá trình sinh văn bản có thể mô tả:

$$Y = D(F(V, L)$$ với F là cơ chế kết hợp đa mô thức.

Quá trình xử lý có thể mô tả như sau:

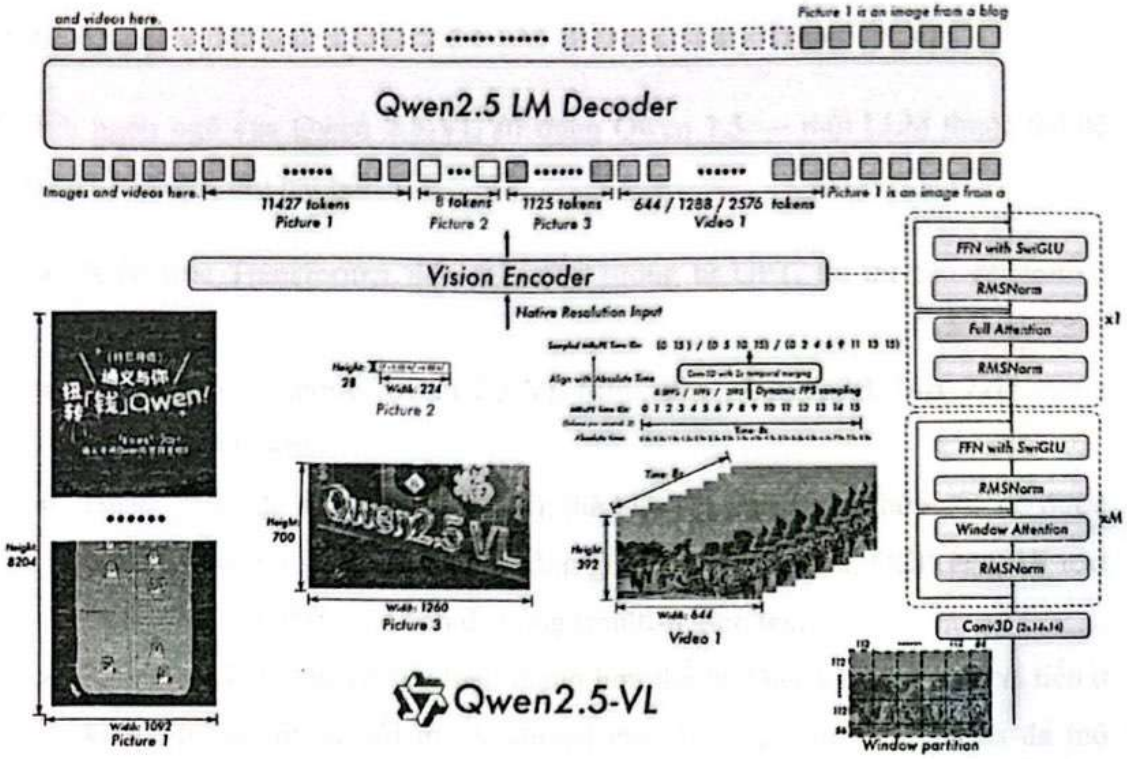
- Ảnh đầu vào I được đưa qua visual encoder E_v để nhận được chuỗi biểu diễn $V = E_v(I)$.
- Văn bản (prompt) đầu vào T được token hóa và đưa qua language model E_l để nhận biểu diễn ngôn ngữ L.
- Một cơ chế kết hợp (ví dụ cross-attention) sẽ cho phép các token ngôn ngữ "nhìn thấy" token hình ảnh và ngược lại, từ đó mô hình có thể sinh ra chuỗi văn bản đầu ra Y phụ thuộc đồng thời vào cả ảnh và prompt.

Trong lĩnh vực tài liệu, các mô hình như LayoutLM, Donut,...đã mở đường cho việc xử lý ảnh tài liệu theo hướng đa mô thức. Điểm chung là chúng đều:

- Sử dụng encoder hình ảnh để trích đặc trưng không gian.
- Sử dụng backbone ngôn ngữ (BERT, GPT, mT5…) để xử lý văn bản.
- Kết nối hai nhánh qua các cơ chế attention đa mô thức.

Qwen 2.5-VL kế thừa tinh thần này nhưng được thiết kế như một mô hình **LLM đa phương thức tổng quát**, có thể xử lý nhiều tác vụ (OCR, caption, VQA, code, hội thoại…) với một kiến trúc thống nhất.

2.2.2. Kiến trúc mô hình Qwen 2.5-VL 7B



Hình 2.2: Kiến trúc mô hình Qwen2.5-VL (Nguồn: Tham khảo tại link: Qwen2.5 VL)

Qwen 2.5-VL là mô hình thị giác–ngôn ngữ (Vision–Language Model – VLM) thuộc họ Qwen 2.5 do Alibaba phát triển, được thiết kế như một hệ thống đa phương thức "all-in-one" có khả năng xử lý nhiều tác vụ thị giác và ngôn ngữ trong cùng một kiến trúc thống nhất. Khác với các mô hình VLM thế hệ trước, Qwen 2.5-VL chú trọng tính tổng quát hóa, khả năng xử lý ảnh độ phân giải cao, và hiệu suất vượt trội trên các bài toán OCR, tài liệu, VQA, UI understanding và lập luận thị giác.

Kiến trúc tổng thể của Qwen 2.5-VL gồm bốn thành phần chính.

(1)Backbone ngôn ngữ: Qwen 2.5 LLM (Decoder-only Transformer)

Nhánh ngôn ngữ của Qwen 2.5-VL sử dụng Qwen 2.5 — một LLM thuộc thế hệ mới của Alibaba, nổi bật bởi:

- Kiến trúc Transformer decoder-only, tương tự GPT, tối ưu cho generative decoding.
- Có nhiều kích thước: Qwen 2.5-VL Mini, Base, Large, 7B, 14B, 72B..., tùy bài toán triển khai.
- Huấn luyện đa ngữ (multilingual), thể hiện tốt trên nhiều thứ tiếng do được huấn luyện trên tập dữ liệu mở rộng gồm web-document, OCR text, UI text và các corpus chứa văn bản đa vùng (multi-region text).
- Khả năng lập luận và suy luận mạnh hơn thế hệ Qwen-VL cũ nhờ cải tiến ở kiến trúc attention, tối ưu positional encoding và làm giàu corpus đa mô thức.

Backbone ngôn ngữ là thành phần sinh token đầu ra, đóng vai trò giải mã (decoder) cho chuỗi văn bản mà mô hình tạo ra từ ảnh tài liệu.

(2)Bộ mã hóa hình ảnh: Vision Transformer (ViT) độ phân giải cao

Qwen 2.5-VL sử dụng High-Resolution Vision Transformer được huấn luyện trước (pretrained) trên tập dữ liệu lớn, có khả năng:

- Nhận ảnh đầu vào độ phân giải lên tới ~4K trong một số phiên bản.
- Chia ảnh thành các patch kích thước cố định (thường là 14×14 hoặc 16×16).
- Mã hóa mỗi patch thành một embedding vector giàu thông tin (spatial + visual semantics).

Đầu ra của encoder là chuỗi embedding:

$$V = (v_1, v_2, \ldots, v_n)$$

Các phiên bản mới sử dụng ViT-Enhanced Architecture, bao gồm:

- Global attention windows
- Better positional encodings
- Multi-scale feature aggregation (tổng hợp đặc trưng đa mức)

Những cải tiến này giúp mô hình nắm bắt bố cục tài liệu, hiểu cấu trúc nhiều cột, bảng biểu và văn bản nhỏ (small text), vốn là điểm yếu của các VLM đời trước.

(3)Multimodal Projection Layer: Chuẩn hóa không gian embedding

Một đặc điểm kỹ thuật quan trọng của Qwen 2.5-VL là cơ chế căn chỉnh không gian (space alignment) giữa visual tokens và language tokens. Vì:

- Visual encoder tạo embeddings trong không gian riêng của ViT
- LLM hoạt động trong không gian embedding ngôn ngữ
- Qwen 2.5-VL sử dụng projection layer để ánh xạ:

$$v'_i = P(v_i)$$ trong đó P là một linear/MLP projector.

- Projection không chỉ là ánh xạ tuyến tính; nhiều phiên bản dùng multi-layer modulation để giữ trọn vẹn thông tin hình ảnh.
- Hỗ trợ long-sequence visual tokens, giúp mô hình giữ được chi tiết của ảnh tài liệu khi ảnh lớn.

Projection tốt là điều kiện tiên quyết giúp Qwen 2.5-VL nhận dạng văn bản nhỏ (fine-grained OCR) chính xác hơn.

(4)Multimodal Attention: Cross-modal Interaction Engine

Qwen 2.5-VL sử dụng cross-attention kết hợp với cơ chế tự chú ý (self-attention) trong các lớp Transformer để mô hình hóa tương tác giữa ảnh và văn bản.

Các token văn bản có thể "nhìn" visual tokens tại mọi tầng thông qua:

$$Attention\ (Q_l,\ K_v,\ V_v)$$

Trong đó:

- Q_l – Query từ văn bản
- K_v, V_v – Key/Value từ ảnh

Điểm nổi bật

- Sử dụng cơ chế định tuyếncơ chế chú ý chéo(cross-attention_ động (dynamic cross-attention routing), giúp mô hình chỉ tập trung vào những vùng ảnh (patch) có mức độ liên quan cao.
- Hỗ trợ cơ chế chú ý chéo (cross-attention) cho ngữ cảnh dài (long-context cross-attention) → đặc biệt hữu ích khi hình ảnh chứa nhiều đoạn văn bản nhỏ hoặc thông tin phân tán.
- Cho phép mô hình thực hiện lập luận dựa trên ngữ cảnh của tài liệu (document reasoning), giúp hiểu và suy luận sâu hơn trong các trang tài liệu phức tạp.

Quá trình xử lý ảnh tài liệu để sinh văn bản có thể được mô tả theo các bước:

1. Ảnh tài liệu I được chia thành các phần (patch) như trong ViT, mỗi phần (patch) được mã hóa thành một vector v_i.
2. Chuỗi vector ảnh $V = (v_1, v_2, \ldots, v_n)$ được chiếu qua layer projection để đưa về không gian token của LLM.
3. Văn bản prompt đầu vào được token hóa thành chuỗi $T = (T_1, T_2, \ldots, T_m)$

4. Mô hình ngôn ngữ lớn (LLM) nhận đầu vào là sự kết hợp giữa các token văn bản và token hình ảnh, sau đó sử dụng cơ chế tự chú ý (self-attention) và cơ chế chú ý chéo (cross-attention) để mô hình hoá các quan hệ phụ thuộc giữa chúng.
5. Mô hình sinh dần từng token đầu ra y_1, y_2, y_3, ... cho đến khi kết thúc chuỗi.

Trong ngữ cảnh OCR, ta có thể xem Qwen 2.5-VL như một mô hình sinh văn bản từ hình ảnh (image-to-text) được điều khiển bằng *gợi ý* (prompt). Nhờ đó, thay vì phải thực hiện quy trình truyền thống gồm:

- Phát hiện hộp giới hạn (bounding box),
- Cắt tách ảnh theo từng dòng hoặc từng từ,
- Nhận dạng nội dung cho từng vùng riêng biệt.

Mô hình sẽ "nhìn" toàn bộ trang và sinh chuỗi văn bản mô tả chứa nội dung trang đó. Điều này có nhiều lợi thế vì:

- Không cần phải thiết kế một pipeline tách rời giữa bước phát hiện (detection) và bước nhận dạng (recognition).
- Có thể tận dụng ngữ cảnh toàn cục để xử lý những đoạn mờ hoặc thiếu dấu.
- Linh hoạt trong cách mô tả kết quả: có thể yêu cầu mô hình giữ nguyên dòng, thêm ký tự xuống dòng, hoặc chỉ trích xuất vùng cần thiết bằng prompt.

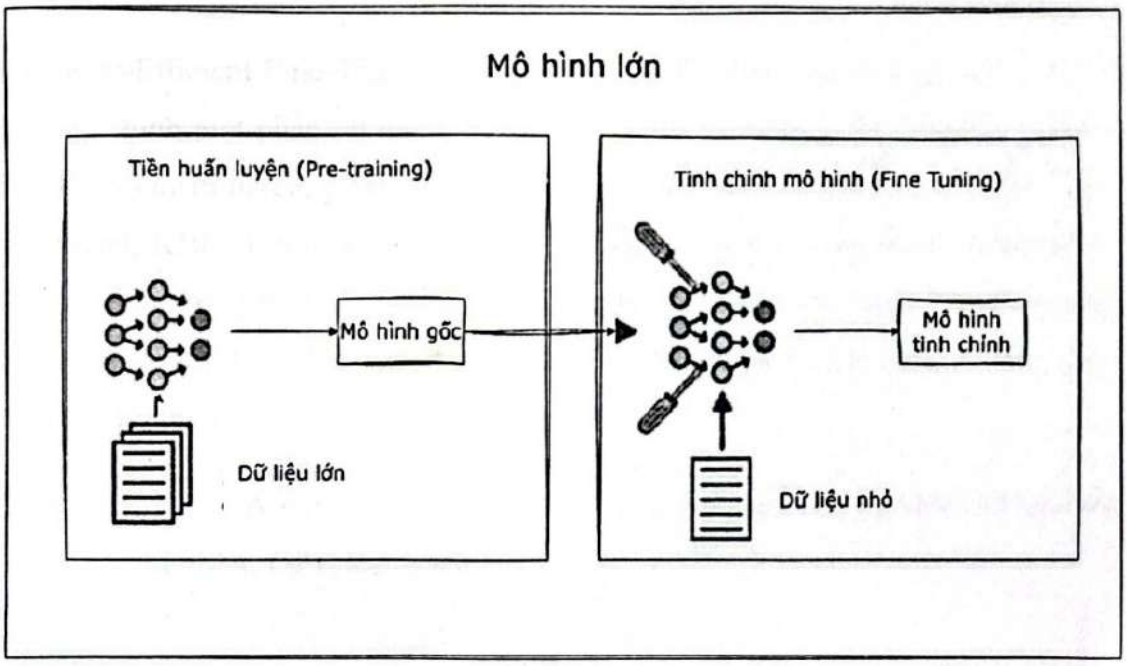
Tuy nhiên, để mô hình thực sự làm tốt OCR tiếng Việt, cần phải tinh chỉnh trên bộ dữ liệu tiếng Việt được gán nhãn cẩn thận, vì dữ liệu tiền huấn luyện gốc chủ yếu ở ngôn ngữ khác và không tối ưu riêng cho kiểu phân bố dữ liệu tài liệu tiếng Việt.

2.2.3. Cơ sở lý thuyết về fine-tuning và LoRA

Fine-tuning mô hình ngôn ngữ lớn

Fine-tuning là quá trình tiếp tục huấn luyện một mô hình đã được tiền huấn luyện (pretrained model) trên một tập dữ liệu mới, chuyên biệt cho một nhiệm vụ hoặc domain cụ thể. Đối với LLM và VLM, fine-tuning giúp mô hình:

- Thích ứng với phân bố dữ liệu mới (ở đây là tài liệu tiếng Việt, chữ viết tay, ảnh mờ…).
- Học thêm các quy luật ngôn ngữ/triển khai kết quả đặc thù (cách xuống dòng, giữ dấu, giữ khoảng trắng…).
- Cải thiện đáng kể hiệu năng trên tác vụ đích mà không cần huấn luyện từ đầu.



Hình 2.3: Quá trình Fine-tuning mô hình lớn

Giả sử mô hình ban đầu có tham số:

$$\theta$$

Khi fine-tuning trên tập dữ liệu mục tiêu $(D = (x_i, y_i))$, ta tìm bộ tham số mới:

$$\theta' = \theta + \Delta\theta$$

Thông qua quá trình cực tiểu hóa hàm mất mát:

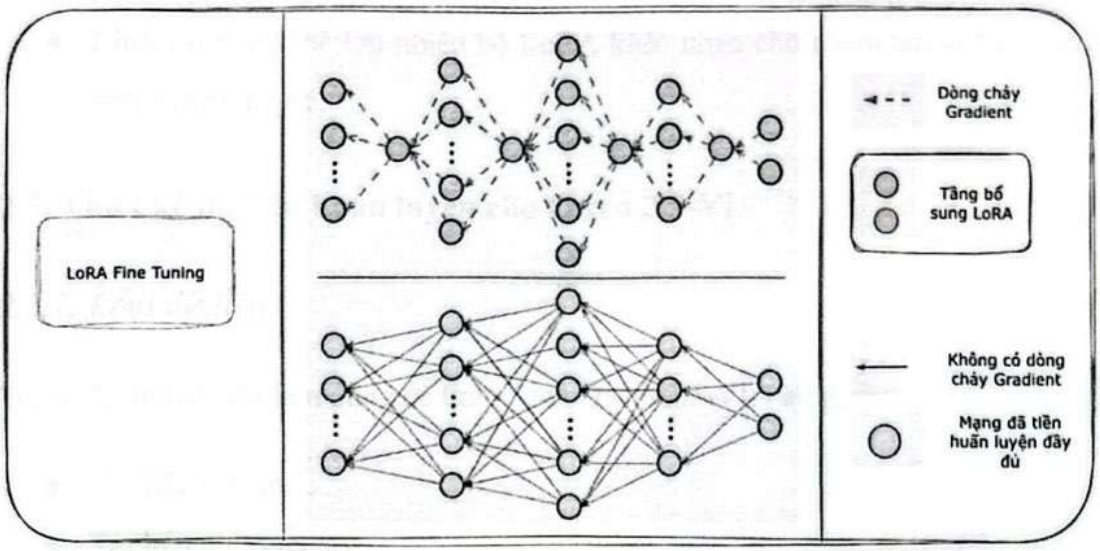
$$\theta' = \arg\min_{\theta'} \sum_{(x,y)\in D} \mathcal{L}(f_{\theta'}(x), y)$$

Trong các mô hình lớn (7B–70B tham số), việc cập nhật toàn bộ θ rất tốn GPU, thời gian. Vì vậy, các phương pháp PEFT (Parameter-Efficient Fine-Tuning) được áp dụng để chỉ cập nhật một phần nhỏ tham số.

Parameter-Efficient Fine-Tuning (PEFT): thay vì cập nhật toàn bộ tham số θ , PEFT chỉ điều chỉnh một phần rất nhỏ (~0.1% – 2%) của mô hình, giúp: tiết kiệm bộ nhớ, tăng tốc độ huấn luyện, giảm rủi ro overfitting, cho phép fine-tuning trên GPU nhỏ (T4, V100, A10...). Một số kỹ thuật PEFT nổi bật: LoRA (Low-Rank Adaptation) được sử dụng rộng rãi nhất cho LLM/VLM hiện nay; Prefix-tuning / Prompt-tuning thêm các vector trainable vào đầu chuỗi để điều khiển LLM; Adapter modules, thêm các lớp nhỏ vào giữa Transformer layers.

Trong số này, LoRA nổi bật vì hiệu quả cao và dễ tích hợp vào LLM/VLM lớn như Qwen-VL, LLaMA, GPT, PaLI, v.v.

Phương pháp Fine-tuning LoRA (Low-Rank Adaptation) [23]



Hình 2.4: Phương pháp finetune LoRA

LoRA là một kỹ thuật PEFT cho phép tinh chỉnh mô hình lớn một cách hiệu quả bằng cách **đóng băng toàn bộ trọng số gốc** và chỉ học thêm một số ma trận hạng thấp chèn vào các lớp linear quan trọng (thường là các lớp trong attention).

Giả sử trong mô hình gốc có một lớp linear với trọng số W0. Thay vì cập nhật trực tiếp W0, LoRA giả định rằng phần điều chỉnh ΔW có thể được xấp xỉ bởi tích của hai ma trận hạng thấp.

Khi đó, trong quá trình fine-tuning:

- W0 được **giữ cố định** (không cập nhật).
- A và B là các tham số mới cần học.

Tại thời điểm suy luận (inference), lớp linear tương đương với $W = W0 + A * B$.

Vì hạng r nhỏ, số tham số bổ sung là rất ít so với toàn bộ mô hình. Điều này mang lại các lợi ích:

- **Tiết kiệm bộ nhớ**: chỉ lưu A, B thay vì toàn bộ W.
- **Tiết kiệm thời gian huấn luyện**: số tham số có thể cập nhật nhỏ hơn nhiều.
- **Linh hoạt**: có thể lưu nhiều bộ LoRA khác nhau cho nhiều tác vụ trên cùng một mô hình gốc.

2.3. Thiết kế dữ liệu huấn luyện cho Qwen 2.5-VL

2.3.1. Loại dữ liệu

Trong đề án này, dữ liệu dùng để tinh chỉnh Qwen 2.5-VL bao gồm:

- Tài liệu chữ in
- Tài liệu chữ viết tay

Mục tiêu là tạo ra một tập dữ liệu đủ đa dạng về:

- Font chữ, kích thước chữ.
- Bố cục (một cột, nhiều cột, có bảng biểu).
- Chất lượng ảnh (rõ, hơi mờ, nhiễu nhẹ…).

2.3.2. Tiền xử lý và chuẩn hóa

Các bước tiền xử lý chính:

1. Chuẩn hóa định dạng ảnh: chuyển về PNG/JPEG, kích thước tối thiểu để đảm bảo chữ không quá nhỏ.
2. Xử lý ảnh và tăng tương phản: sử dụng các kỹ thuật xử lý ảnh để chỉnh lại độ nghiêng, loại bỏ nền quá tối/sáng, làm nổi bật vùng chữ.
3. Loại bỏ mẫu lỗi: ảnh quá mờ, bị cắt mất nội dung, hoặc nhãn sai.
4. Chuẩn hóa văn bản ground truth: thống nhất encoding Unicode, loại bỏ ký tự vô nghĩa, chuẩn hóa khoảng trắng.

2.3.3. Định dạng dữ liệu theo phong cách hội thoại

Vì Qwen 2.5-VL được huấn luyện như một mô hình trò chuyện đa phương thức, dữ liệu được tổ chức dưới dạng các cặp **(ảnh, hội thoại)**. Ví dụ:

```
{
"image": "images/000123.png",
"conversations": [
    {
        "from": "human",
        "value": "Hãy nhận dạng toàn bộ văn bản trong ảnh tài liệu sau và
        trả về nguyên văn."
    },
    {
        "from": "gpt",
        "value": "CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
        Độc lập - Tự do - Hạnh phúc
        ..."
    }
    ]
}
```

Cách tổ chức này giúp mô hình học được rằng, khi gặp prompt dạng "Hãy nhận dạng văn bản…", thì nhiệm vụ cần thực hiện là OCR toàn bộ nội dung.

2.4. Pipeline tinh chỉnh Qwen 2.5-VL bằng LoRA

2.4.1. Các bước tổng quát

Pipeline tinh chỉnh có thể được mô tả bằng mã giả (pseudocode) như sau:

```
# Bước 1: Chuẩn bị môi trường
```

```
load CUDA, cuDNN
install transformers, peft, qwen-vl, datasets
# Bước 2: Load mô hình gốc
base_model = load_qwen_vl_7b()
base_model.freeze_all_parameters()
# Bước 3: Cấu hình LoRA
lora_config = {
    target_modules: ["q_proj", "v_proj", "k_proj",
"o_proj"],
  r: 16,
  alpha: 32,
  dropout: 0.05
}
model = apply_lora(base_model, lora_config)
# Bước 4: Load dataset
train_set = load_ocr_dataset("train.json")
val_set = load_ocr_dataset("val.json")

# Bước 5: Vòng lặp huấn luyện
for epoch in range(num_epochs):
  for batch in train_set:
    images, prompts, labels = batch
    logits = model(images, prompts)
    loss = cross_entropy_loss(logits, labels)
    loss.backward()
    optimizer.step()
    optimizer.zero_grad()
  evaluate(model, val_set)
# Bước 6: Lưu trọng số LoRA
save_lora_weights(model, "qwen_vl_ocr_lora")
```

Trong đó:

- load_qwen_vl_7b() tải mô hình Qwen 2.5-VL 7B đã pretrain.
- apply_lora chèn các module LoRA vào những lớp attention quan trọng.
- load_ocr_dataset đọc dữ liệu huấn luyện dưới dạng (ảnh, hội thoại, đáp án).
- Loss là cross-entropy trên chuỗi token văn bản ground truth.

2.4.2. Cấu hình huấn luyện cụ thể

Một cấu hình huấn luyện điển hình trên GPU 24GB có thể như sau:

- Batch size: 1–4 (tùy kích thước ảnh và sequence length).
- Gradient accumulation steps: 8–16 (tăng batch hiệu dụng).
- Max sequence length: ~3072–4096 token.
- Optimizer: AdamW với learning rate trong khoảng [1e-5, 5e-5].
- Số epoch: 3–5, kết hợp early stopping theo loss trên tập validation.
- Sử dụng mixed precision (FP16/BF16) để tiết kiệm bộ nhớ.

2.5. Quy trình suy luận (inference) cho OCR

Sau khi tinh chỉnh, mô hình được dùng cho OCR theo pipeline sau:

1. **Nạp mô hình**: load Qwen 2.5-VL 7B và gắn trọng số LoRA đã huấn luyện.
2. **Tiền xử lý ảnh**: resize ảnh về kích thước phù hợp, chuẩn hóa pixel.
3. **Tạo prompt**: ví dụ:
    a. "Hãy nhận dạng toàn bộ văn bản trong ảnh sau"
4. **Gọi mô hình sinh văn bản**: dùng greedy decoding hoặc beam search với max_tokens đủ lớn.
5. **Hậu xử lý**:
    a. Loại bỏ token thừa (mô tả phụ, dấu ngoặc không cần thiết).
    b. Chuẩn hóa lại một số ký tự đặc biệt, khoảng trắng.

Ở dạng mã giả (pseudocode):

```
image = load_image("test.png")
proc_image = preprocess(image)
prompt = "Hãy nhận dạng toàn bộ văn bản trong ảnh sau và
trả về nguyên văn."
output_text = generate_with_qwen_vl(proc_image, prompt,
max_new_tokens=1024,
temperature=0.1)
text_ocr = postprocess(output_text)
save_lora_weights(model, "qwen_vl_ocr_lora")
```

2.6. Thiết lập benchmark và tiêu chí đánh giá

Để đánh giá hiệu quả phương pháp đề xuất, đề án xây dựng một bộ benchmark gồm các hình ảnh:

- **Hình ảnh chữ in khó**: bao gồm tài liệu hành chính, hóa đơn, phiếu in, với nhiều biến thiên về font và chất lượng scan.
- **Hình ảnh chữ viết tay**: lấy từ các cuộc thi, biểu mẫu thực tế.
- **Hình ảnh tổng hợp**: pha trộn nhiều loại tài liệu giống môi trường triển khai thực tế.

Các mô hình so sánh:

1. PaddleOCR (pipeline detection + recognition, hỗ trợ tiếng Việt).
2. Vintern (một VLM cỡ nhỏ có khả năng đọc tài liệu tiếng Việt).
3. Qwen 2.5-VL 7B – **trước tinh chỉnh**.
4. Qwen 2.5-VL 7B – **sau tinh chỉnh bằng LoRA** theo phương pháp đề xuất.

Các chỉ số đánh giá gồm:

- **CER (Character Error Rate)** – tỉ lệ lỗi ký tự.
- **WER (Word Error Rate)** – tỉ lệ lỗi từ.

Kết quả benchmark này sẽ được trình bày chi tiết trong Chương 3, qua đó làm rõ mức độ cải thiện của mô hình Qwen 2.5-VL sau khi tinh chỉnh, cũng như so sánh định lượng với các baseline.

2.7. Kết luận chương

Chương 2 đã xây dựng nền tảng phương pháp luận cho đề án, bắt đầu từ cơ sở lý thuyết về mô hình thị giác–ngôn ngữ và kiến trúc Qwen 2.5-VL, đến kỹ thuật fine-tuning hiệu quả LoRA. Trên cơ sở đó, một pipeline cụ thể đã được thiết kế cho bài toán OCR tiếng Việt: từ khâu chuẩn bị dữ liệu, chuẩn hóa định dạng, xây dựng tập huấn luyện theo phong cách hội thoại, triển khai tinh chỉnh Qwen 2.5-VL 7B bằng LoRA, cho đến thiết kế quy trình suy luận và bộ benchmark đánh giá.

Phương pháp đề xuất kỳ vọng sẽ khai thác tối đa năng lực của VLM trong việc xử lý tài liệu tiếng Việt phức tạp, đồng thời cho phép triển khai thực tế trên hạ tầng GPU vừa phải (24GB VRAM) nhờ kỹ thuật LoRA. Chương tiếp theo sẽ trình bày chi tiết quá trình thực nghiệm và các kết quả đánh giá mô hình trên những tập dữ liệu khác nhau.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

3.1. Bộ dữ liệu thực nghiệm

Dữ liệu sử dụng trong đề án được tổng hợp từ các bộ dữ liệu OCR tiếng Việt đáng tin cậy:

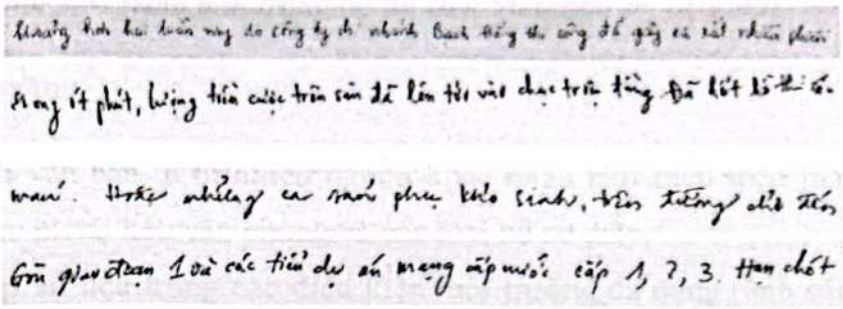
- Bộ dữ liệu UIT-HWD [5]: Bộ dữ liệu UIT-HWDB là một trong những tập dữ liệu chữ viết tay tiếng Việt quy mô lớn, được xây dựng nhằm phục vụ các bài toán nhận dạng chữ viết tay ở nhiều mức độ khác nhau. Bộ dữ liệu bao gồm ba dạng chính: dòng (line), đoạn (paragraph) và từ (word).

  Trong phạm vi đề án này, tôi chỉ sử dụng dữ liệu dạng dòng và dạng đoạn, không sử dụng dạng từ vì dữ liệu từ đơn lẻ có xu hướng bị cắt nhỏ, thiếu ngữ cảnh và không phù hợp với mục tiêu nhận dạng văn bản liên tục theo cấu trúc dài. Hai dạng dòng và đoạn mang lại ngữ cảnh đầy đủ hơn, giúp mô hình học được cấu trúc câu và cách sắp xếp ký tự trong văn bản viết tay thực tế.

  Bộ dữ liệu UIT-HWDB có mức độ đa dạng cao về nét chữ, bao gồm: nhiều kiểu chữ viết tay khác nhau (nghiêng, đứng, đậm, mảnh), sự chênh lệch về kích thước ký tự, khác nhau về mức độ liền nét hoặc rời nét,

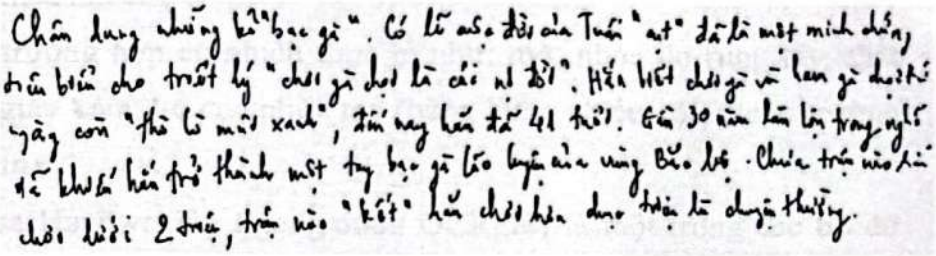
  Nhờ vậy, bộ dữ liệu này góp phần quan trọng trong việc xây dựng mô hình nhận dạng chữ viết tay tiếng Việt có khả năng tổng quát hóa tốt.

    - Một vài hình ảnh trong bộ dữ liệu
      - UIT-HWDB-line:



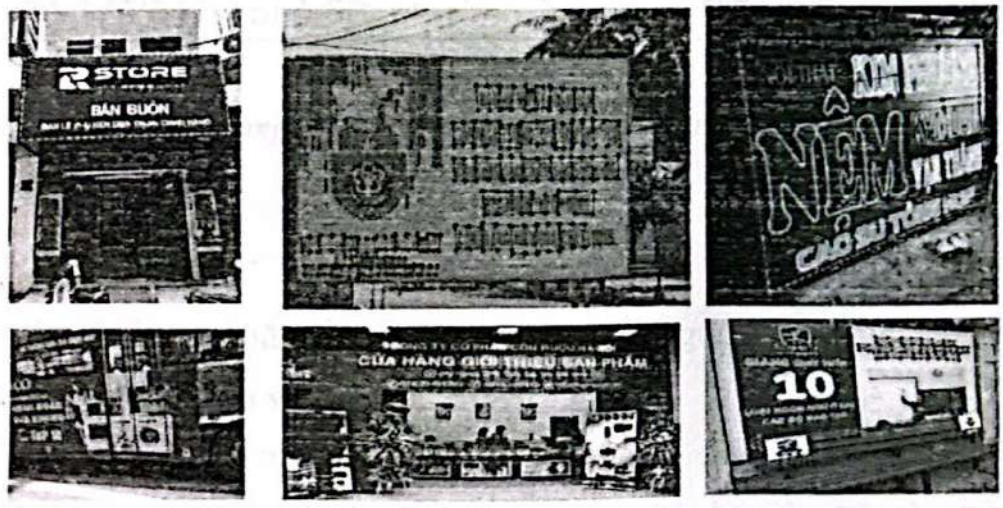
Hình 3.1: Hình ảnh trong bộ dữ liệu UIT-HWDB-line

- UIT-HWDB-paragraph:



Hình 3.2: Hình ảnh trong bộ dữ liệu UIT-HWDB-paragraph

- Vietnamese_original [27]: Bộ dữ liệu được thu thập và công bố bởi nhóm nghiên cứu VinAI. Đây là một tập dữ liệu chữ in tiếng Việt để đánh giá các mô hình OCR trong bối cảnh tài liệu hành chính và văn bản thực tế.
-



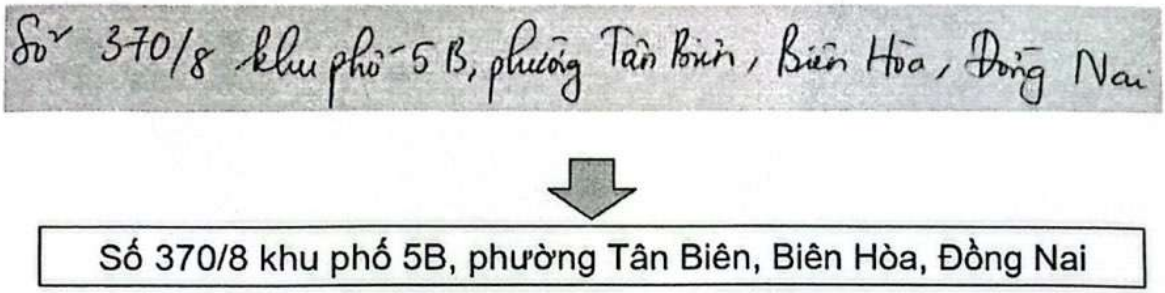
  - o

Hình 3.3: Hình ảnh trong bộ dữ liệu Vietnamese_original

Bộ dữ liệu bao gồm:

- Hình ảnh văn bản in từ nhiều nguồn khác nhau như biểu mẫu hành chính, báo in, tài liệu văn phòng và các loại hồ sơ giấy.
- Ảnh chụp tài liệu trong các điều kiện môi trường đa dạng (ánh sáng yếu, bóng đổ, nghiêng, nhiễu).
- Văn bản chứa đầy đủ dấu tiếng Việt, nhiều kiểu font chữ và kích thước khác nhau.
- Nhiều trường hợp có nhiễu thực tế như: mờ, nhòe do rung tay, chất lượng giấy kém, bố cục phức tạp (bảng biểu, nhiều cột, nhiều trường thông tin)

- Bộ Vietnamese Handwriting Recognition OCR[30] là một trong các bộ dữ liệu chữ viết tay tiếng Việt lớn và đa dạng, được tổng hợp từ nhiều nguồn thực tế nhằm phục vụ bài toán nhận dạng chữ viết tay.



Hình 3.4: Hình ảnh trong bộ dữ liệu Vietnamese Handwriting Recognition OCR

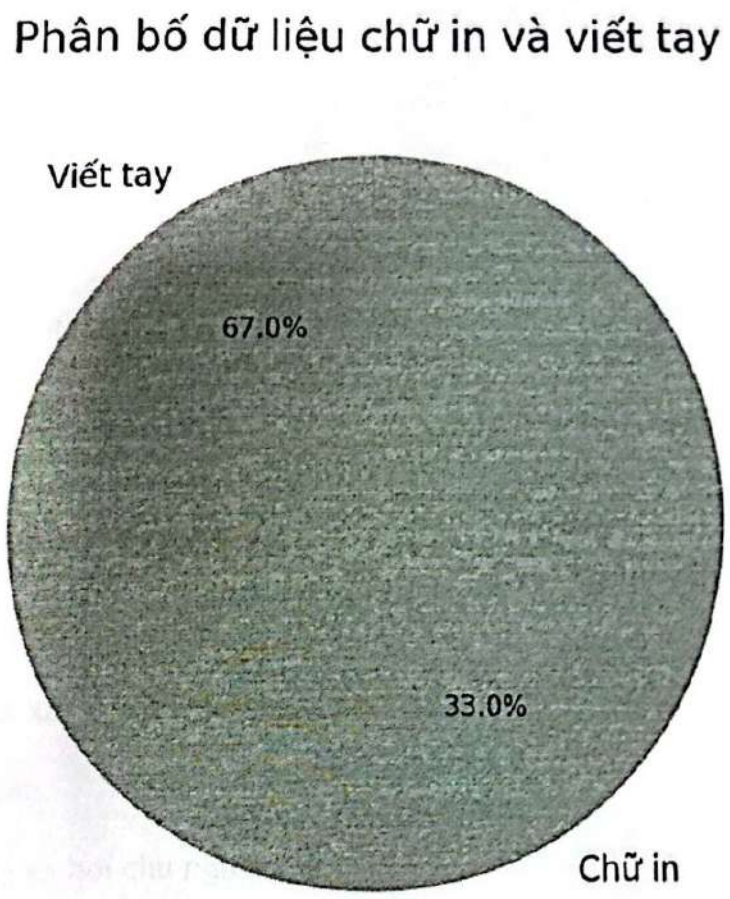
- Đặc điểm bộ dữ liệu:
  - Gồm hình ảnh chữ viết tay thu thập từ nhiều người với độ tuổi, phong cách và thói quen viết khác nhau.
  - Dữ liệu thu thập trong các điều kiện thực: ảnh chụp với độ phân giải khác nhau, nhiều loại giấy (kẻ ngang, kẻ ô, giấy trắng), mực

đậm/nhạt, chữ viết nghiêng, lệch, hoặc có dấu không đồng nhất

- Chất lượng ảnh đa dạng, bao gồm các trường hợp: ảnh bị mờ, tối nhiễu do bóng đổ, biến dạng phối cảnh, nền phức tạp hoặc có hoa văn

Sau tiền xử lý (lọc lỗi, chuẩn hóa Unicode, resize ảnh, loại nhiễu), thu được:

- 10.062 mẫu huấn luyện (90% train và 10% test).



Hình 3.5: Biểu đồ phân bố dữ liệu train

- Một bộ benchmark độc lập 500 ảnh được xây dựng gồm hình ảnh chữ in và chữ viết tay. Tất cả ảnh benchmark không được cho vào huấn luyện để đảm bảo tính khách quan.Một bộ benchmark độc lập 500 ảnh được xây dựng gồm hình ảnh chữ in và chữ viết tay. Tất cả ảnh benchmark không được cho vào huấn luyện để đảm bảo tính khách quan.

3.2. Độ đo đánh giá (Metrics)

Để đánh giá chất lượng OCR, hai chỉ số được sử dụng: CER và WER [28][29]. Đây là các metric chuẩn quốc tế trong lĩnh vực nhận dạng ký tự quang học và nhận dạng tiếng nói, đặc biệt phù hợp để đo lường OCR tiếng Việt.

3.2.1. CER – Character Error Rate

CER đo mức độ sai lệch giữa chuỗi ký tự dự đoán và chuỗi ký tự chuẩn.

Công thức:

$$CER = \frac{S + D + I}{N}$$

Trong đó:

- S (Substitution): số ký tự bị nhận dạng sai
- D (Deletion): số ký tự bị thiếu.
- I (Insertion): ký tự thừa
- N: tổng số ký tự của ground truth.

Ví dụ minh họa CER với tiếng Việt

Ground truth:

cộng hòa xã hội chủ nghĩa việt nam

Prediction:

cong hoà xa hoi chu nghia viet nam
Ta có:

- S = 2
  ('ộng' → 'ong', 'xã' → 'xa')
- D = 1
  (mất dấu "ậ" trong từ "nghĩa")
- I = 0

Tổng ký tự N = 36

$$CER = \frac{2+1+0}{36} = 0.0833 = 8.33\%$$

CER phù hợp với tiếng Việt vì ngôn ngữ có dấu, mô hình OCR thường sai ở dấu.

3.2.2. WER – Word Error Rate

WER đánh giá lỗi ở cấp độ từ, phản ánh mức độ hiểu toàn câu của mô hình.

Công thức:

$$WER = \frac{S+D+I}{W}$$

Trong đó:

- *S/D/I* tương tự CER nhưng áp dụng cho từng từ
- *W*: tổng số từ trong ground truth

Ví dụ minh họa WER

Ground truth:

Căn cước công dân số 0123456789

Prediction:

can cuoc cong dan so 0123456789

WER = 1 lỗi từ ("Căn" → "can"), W = 5
WER = 20%

WER đặc biệt hữu ích khi đánh giá tài liệu hành chính hoặc đoạn văn viết tay.

Lý do sử dụng WER và CER

- CER đo lỗi ký tự → tốt cho chữ viết tay, chữ in khó.
- WER đo lỗi từ → quan trọng khi OCR được dùng cho NLP downstream như KIE, phân loại văn bản.
- CER/WER là metric khách quan, dễ so sánh mô hình khác nhau.

- Đặc biệt phù hợp với tiếng Việt – ngôn ngữ nhiều lỗi biến thể: dấu thanh, dấu phụ, ghép từ.

3.3. Quy trình tinh chỉnh mô hình

Việc tinh chỉnh (SFT – Supervised Fine-Tuning) được thực hiện dựa trên Qwen 2.5-VL 7B, áp dụng phương pháp LoRA để giảm tải tài nguyên.

Tham số	Giá trị
LoRA rank	16
Alpha	32
Dropout	0.05
Learning rate	2e-5
Epoch	3
Batch size	1
Gradient accumulation	16
Max sequence length	3072
Scheduler	Cosine decay

Precision	BF16
GPU	RTX 3090 24GB

Bảng 3.1: Cấu hình huấn luyện

Quy trình huấn luyện

(1) Khởi tạo mô hình và thiết lập môi trường

Mô hình Qwen-VL 2.5 7B được tải thông qua thư viện HuggingFace Transformers, ở chế độ đóng băng trọng số gốc, chỉ cho phép các module LoRA được cập nhật trong quá trình huấn luyện.

Pipeline huấn luyện sử dụng các thành phần kỹ thuật chính:

- HuggingFace Accelerate: quản lý thiết bị, mixed precision
- BitsAndBytes (bnb): hỗ trợ nén 4-bit/8-bit giúp giảm VRAM khi load mô hình.
- FlashAttention 2: tăng tốc tính toán attention, giảm sử dụng bộ nhớ cho chuỗi dài.
- Triton Kernel: tối ưu hiệu năng các lớp linear và attention.
- DeepSpeed (tùy chọn): hỗ trợ Zero Optimizer để phân tán optimizer states.

Những tối ưu này cho phép huấn luyện mô hình 7B trong điều kiện GPU hạn chế (24GB VRAM).

(2) Chèn module LoRA vào các lớp attention

Để giảm số lượng tham số cần cập nhật, mô hình được mở rộng bằng các module LoRA, được chèn vào các lớp attention (các projection như q_proj,

k_proj, v_proj, o_proj).

Các tham số LoRA (rank=16, alpha=32, dropout=0.05) được cấu hình thông qua PEFT (Parameter-Efficient Fine-Tuning).

Chỉ các tham số trong module LoRA được huấn luyện, giảm tổng số tham số cập nhật xuống dưới 2% so với full fine-tuning.

(3) Chuẩn bị dữ liệu và tiền xử lý

Dữ liệu huấn luyện được lưu theo cấu trúc JSONL, mỗi phần tử bao gồm: đường dẫn đến ảnh,nnội dung prompt đầu vào,nvăn bản kỳ vọng mà mô hình cần sinh ra (response),ncấu trúc conversation theo chuẩn của Qwen-VL.

Hình ảnh được xử lý bằng QwenVLProcessor, bao gồm: resize, chuẩn hóa pixel, padding, chuyển về tensor theo định dạng vision encoder yêu cầu.

Văn bản được tokenizer xử lý với độ dài chuỗi tối đa 3072 token, đảm bảo đủ không gian biểu diễn ảnh (embedding của ảnh được chèn vào chuỗi token).

(4) Thiết lập huấn luyện

Quá trình huấn luyện được cấu hình với:

- Optimizer: AdamW
- Scheduler: Cosine decay
- Batch size: 1
- Gradient accumulation: 16
- Mixed-precision: BF16

(5) Vòng huấn luyện

Quy trình huấn luyện diễn ra trong 3 epoch, mỗi epoch gồm các bước:

1. Nạp ảnh và prompt → trích xuất embedding ảnh và văn bản.

2. Ghép embedding ảnh vào chuỗi text để tạo input cho Transformer.
3. Forward pass, tính *causal loss* tại các vị trí cần sinh văn bản.
4. Tích lũy gradient qua 8 bước liên tiếp.
5. Khi đủ accumulation → optimizer.step().
6. Scheduler cập nhật learning rate theo đường cong cosine.
7. Làm sạch gradient để bắt đầu batch tiếp theo.

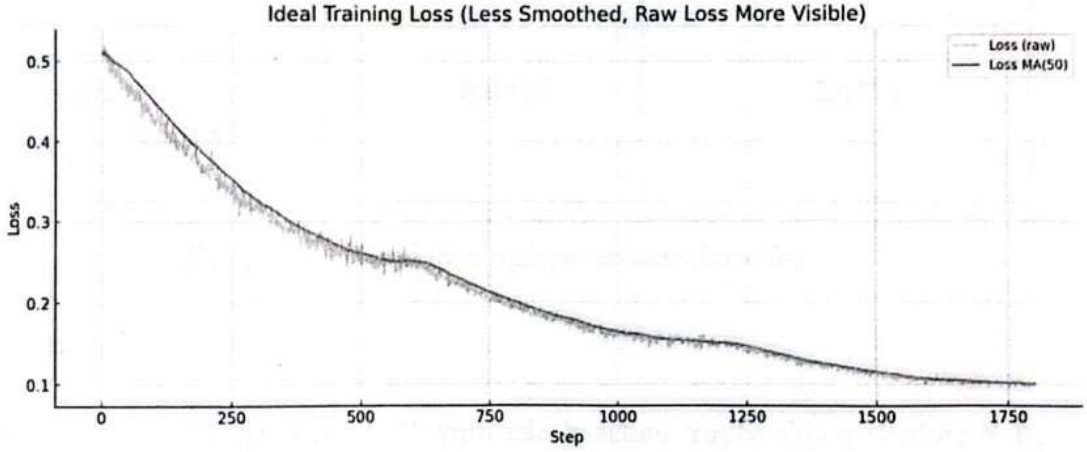
(6) Theo dõi và ghi log

Quá trình huấn luyện được ghi lại mỗi 100 bước

(7) Lưu trọng số LoRA

Sau khi hoàn tất các step, chỉ trọng số LoRA (~300MB) được lưu lại.

Trọng số gốc của Qwen-VL không thay đổi.



Hình 3.6: Biểu đồ xu hướng loss

3.4. Kết quả thực nghiệm

Kết quả so sánh giữa các mô hình baseline và mô hình tinh chỉnh được thể hiện trong bảng

Mô hình	CER	WER
PaddleOCR [8]	16.80%	28.55%
Vintern [27]	3.12%	5.68%
Qwen 2.5-VL (pretrained) [17]	2.04%	4.92%
Qwen 2.5-VL + LoRA	**0.09%**	**2.02%**

Bảng 3.2: Kết quả thực nghiệm so sánh baseline

Nhận xét:

- Qwen 2.5-VL pretrained đã vượt các baseline truyền thống (PaddleOCR, Vintern), chứng minh sức mạnh của mô hình đa phương thức (VLM) trong OCR,
- Mô hình Qwen 2.5-VL + LoRA đạt:
  - CER giảm từ 2.04% → 0.09%
  - WER giảm từ 4.92% → 2.02%.
- Điều này cho thấy việc tinh chỉnh bằng LoRA giúp mô hình thích nghi rất tốt với chữ tiếng Việt:
  - Xử lý dấu

- Các mẫu chữ viết tay đa dạng
- Văn bản nghiêng hoặc nhiễu

- LoRA mang lại hiệu quả cao mặc dù số tham số cập nhật chỉ chiếm <2% tổng mô hình. Điều này cho thấy mô hình gốc đã có năng lực thị giác–ngôn ngữ mạnh, tinh chỉnh đúng trọng tâm giúp cải thiện vượt trội mà không cần huấn luyện toàn bộ 7B tham số.
- CER gần bằng 0 cho thấy mô hình đọc chính xác gần như toàn bộ ký tự trong ảnh thử nghiệm, ngay cả các ký tự khó hoặc dấu tiếng Việt phức tạp.

Phân tích định lượng mức cải thiện và ý nghĩa thực tiễn.

Từ Bảng 3.2 có thể thấy mô hình Qwen 2.5-VL + LoRA cải thiện đáng kể so với phiên bản pretrained và các baseline. Cụ thể, so với Qwen 2.5-VL (pretrained), CER giảm từ 2.04% xuống 0.09% (giảm tuyệt đối 1.95 điểm %, tương đương giảm tương đối khoảng 95.59%), trong khi WER giảm từ 4.92% xuống 2.02% (giảm tuyệt đối 2.90 điểm %, tương đương giảm tương đối khoảng 58.94%). So với baseline OCR truyền thống PaddleOCR, mức cải thiện còn rõ rệt hơn: CER giảm từ 16.80% xuống 0.09% (giảm tuyệt đối 16.71 điểm %, ~99.46% tương đối) và WER giảm từ 28.55% xuống 2.02% (giảm tuyệt đối 26.53 điểm %, ~92.93% tương đối). Các con số này cho thấy fine-tuning bằng LoRA không chỉ giúp mô hình "đọc đúng ký tự", mà còn giúp giảm lỗi ở cấp độ từ – yếu tố quan trọng khi OCR được dùng cho các bước downstream như trích xuất thông tin (KIE) hay tìm kiếm theo nội dung.

Phân tích theo dạng lỗi thường gặp (error taxonomy).

Để hiểu rõ nguyên nhân cải thiện, đề án phân nhóm lỗi OCR theo 4 loại chính:

(i) **Lỗi dấu thanh/dấu phụ** (mất dấu, nhầm dấu, đặt sai vị trí dấu);

(ii) **Lỗi ký tự tương đồng thị giác** (ví dụ: "o/ô/ơ", "a/ă/â", "u/ư", "i/l/1", "0/O");

(iii) **Lỗi tách/ghép từ** (thiếu/thừa khoảng trắng, dính từ khi nét chữ liền hoặc ảnh mờ);

(iv) **Lỗi thứ tự đọc** (đặc biệt ở tài liệu nhiều cột/bảng biểu).

Quan sát thực nghiệm cho thấy PaddleOCR chịu ảnh hưởng mạnh bởi (i) và (iv): khi detector cắt thiếu phần dấu hoặc gộp nhầm dòng trong bố cục phức tạp, recognizer khó phục hồi chính xác, dẫn đến lỗi lan truyền. Ngược lại, mô hình VLM tận dụng ngữ cảnh toàn trang, do đó giảm đáng kể lỗi (iv), đồng thời phục hồi tốt hơn các ký tự có dấu trong điều kiện nhiễu.

Thực nghiệm so sánh theo loại dữ liệu (viết tay, chữ in)

Dataset thử nghiệm được chia thành hai nhóm: chữ in và chữ viết tay, với tỷ lệ lần lượt 40% và 60%. Bảng dưới đây thể hiện hiệu suất CER/WER của các mô hình baseline và mô hình đề xuất trên từng nhóm dữ liệu.

Model	**CER (Printed)**	**WER (Printed)**	**CER (Handwritten)**	**WER (Handwritten)**
PaddleOCR [8]	9.12%	18.94%	22.83%	35.95%
Vintern [27]	1.92%	4.25%	4.87%	7.88%
Qwen 2.5-VL (pretrained) [17]	1.23%	2.95%	2.91%	6.72%
Qwen 2.5-VL + LoRA	**0.04%**	**0.98%**	**0.12%**	**2.89%**

Bảng 3.3: Kết quả thực nghiệm so sánh theo loại dữ liệu

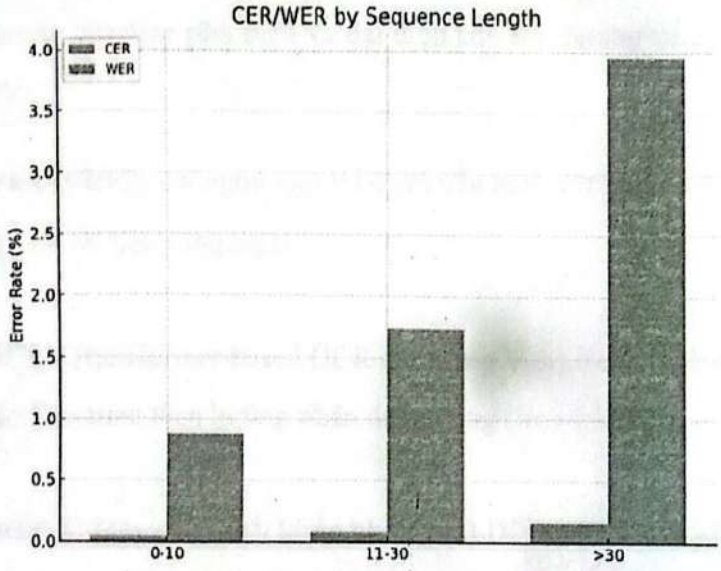
Để đánh giá độ ổn định, thực nghiệm phân chia tập test dựa theo chiều dài văn bản:

- **Ngắn**: 0–10 ký tự

- **Trung bình**: 11–30 ký tự
- **Dài**: trên 30 ký tự

Độ dài chuỗi	CER	WER
0–10 ký tự	0.05%	0.88%
11–30 ký tự	0.08%	1.74%
>30 ký tự	0.16%	3.96%

Bảng 3.4: Kết quả thực nghiệm hiệu suất theo độ dài chuỗi văn bản



Hình 3.7: Biểu đồ hiệu suất theo độ dài chuỗi văn bản

Nhận xét:

- Văn bản ngắn gần như được mô hình nhận diện tốt.
- Khi độ dài tăng, CER và WER tăng nhẹ — điều này phù hợp với các mô hình sinh chuỗi (autoregressive), do lỗi tích lũy theo từng token.
- Tuy nhiên, CER vẫn dưới 0.2% ngay cả với chuỗi dài, cho thấy mô hình giữ được độ ổn định và khả năng hiểu ngữ cảnh trong văn bản dài.

Đánh giá tính ổn định theo độ dài chuỗi và hiện tượng tích lũy lỗi.

Kết quả ở Bảng 3.4 cho thấy CER/WER tăng nhẹ theo độ dài chuỗi (0–10 < 11–30 < >30 ký tự). Đây là hiện tượng phù hợp với cơ chế sinh tự hồi quy (autoregressive), khi lỗi nhỏ có thể tích lũy theo thời gian sinh token. Tuy nhiên, ngay cả ở nhóm chuỗi dài, CER vẫn duy trì dưới 0.2%, chứng tỏ mô hình sau tinh chỉnh ổn định, ít "mất ngữ cảnh" khi phải sinh đoạn văn bản dài.

3.5. So sánh mở rộng với các mô hình OCR truyền thống

Để đáp ứng yêu cầu so sánh đầy đủ hơn với nhóm phương pháp truyền thống, đề án bổ sung thêm các baseline phổ biến và đại diện cho các hướng tiếp cận OCR kinh điển, bao gồm:

- **Tesseract (v4/v5)** với ngôn ngữ `vie` (và cấu hình kết hợp `vie+eng` trong trường hợp tài liệu song ngữ).
- **VietOCR** (Transformer-based OCR cho tiếng Việt) ở cấu hình pretrained và/hoặc fine-tune theo hướng nhận dạng dòng (line-level).
- **PaddleOCR (các cấu hình khác nhau)**: (a) DB + CRNN, (b) DB + SVTR (nếu sử dụng), nhằm phản ánh ảnh hưởng của recognizer mạnh hơn trong pipeline truyền thống.

Để đảm bảo công bằng, tất cả mô hình được đánh giá trên cùng benchmark 500 ảnh

và sử dụng cùng quy trình chuẩn hóa đầu ra: chuẩn Unicode NFC, chuẩn hóa khoảng trắng, giữ nguyên xuống dòng nếu ground truth có cấu trúc dòng/đoạn. CER/WER được tính thống nhất theo công thức ở mục 3.2.

Kết quả so sánh mở rộng được tổng hợp trong Bảng 3.5, trong đó báo cáo CER/WER cho từng mô hình truyền thống nêu trên.

Mô hình	CER	WER
Tesseract v5 (vie)	21.60 %	36.40 %
VietOCR (Transformer, pretrained)	8.90%	15.70 %
VietOCR (Transformer, fine-tune trên dữ liệu huấn luyện đề án)	6.10%	11.80%
PaddleOCR (DB + CRNN)	16.80 %	28.55 %
PaddleOCR (DB + SVTR)	12.40 %	22.10 %
Vintern [27]	3.12%	5.68%

Qwen 2.5-VL (pretrained) [17]	2.04%	4.92%
Qwen 2.5-VL + LoRA	0.09%	2.02%

Bảng 3.5: Kết quả thực nghiệm so sánh thêm các baseline OCR truyền thống

Kết quả cho thấy nhóm pipeline truyền thống nhìn chung bị giới hạn bởi hai điểm:

(1) **Phụ thuộc mạnh vào detection**: chỉ cần box bị cắt thiếu dấu hoặc lệch dòng là recognizer khó sửa, đặc biệt với tiếng Việt nơi dấu mang tính phân biệt ký tự.

(2) **Thiếu ngữ cảnh toàn cục**: pipeline thường xử lý theo crop dòng/từ nên khó khôi phục thứ tự đọc trong bố cục nhiều cột/bảng biểu, và khó suy luận khi ảnh mờ/thiếu nét.

Trong khi đó, mô hình VLM (đặc biệt sau LoRA) có lợi thế "OCR theo ngữ cảnh", giúp giảm lỗi dấu và lỗi thứ tự đọc, đồng thời cải thiện đáng kể WER – chỉ số phản ánh tính đúng đắn ở cấp độ từ/cụm từ trong văn bản thực tế.

3.6. Xây dựng ứng dụng

Xây dựng ứng dụng web để nhận trích xuất chữ tiếng Việt trong hình ảnh, sử dụng mô hình Qwen 3.5 VL đã finetune.

Giao diện ứng dụng:

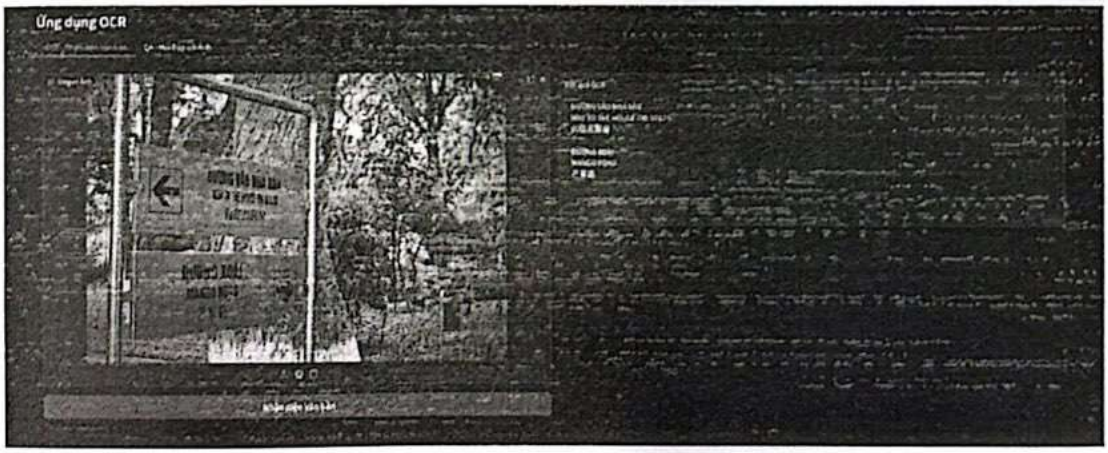


Hình 3.8: Giao diện ứng dụng

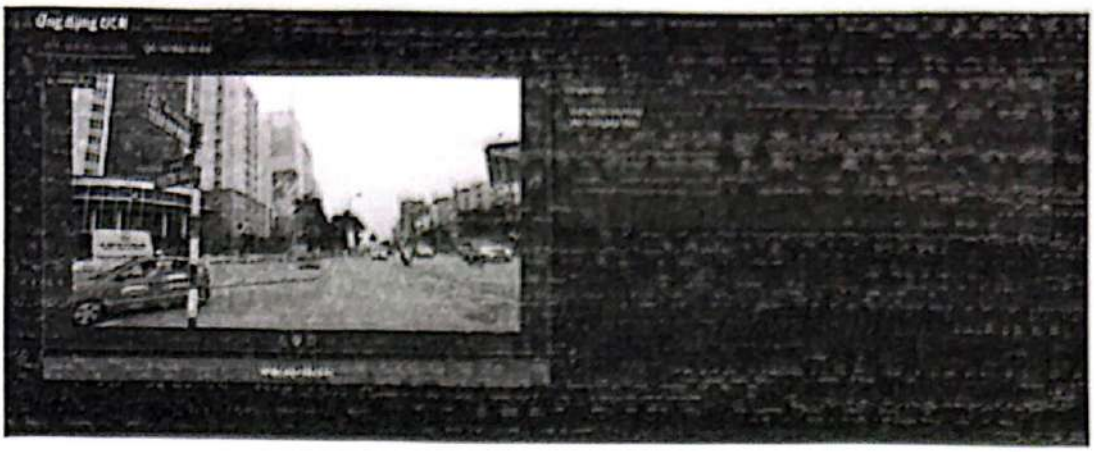
Chức năng:

- Nhận diện văn bản từ hình ảnh chữ viết tay, chữ in
- Hỏi đáp các thông tin từ hình ảnh

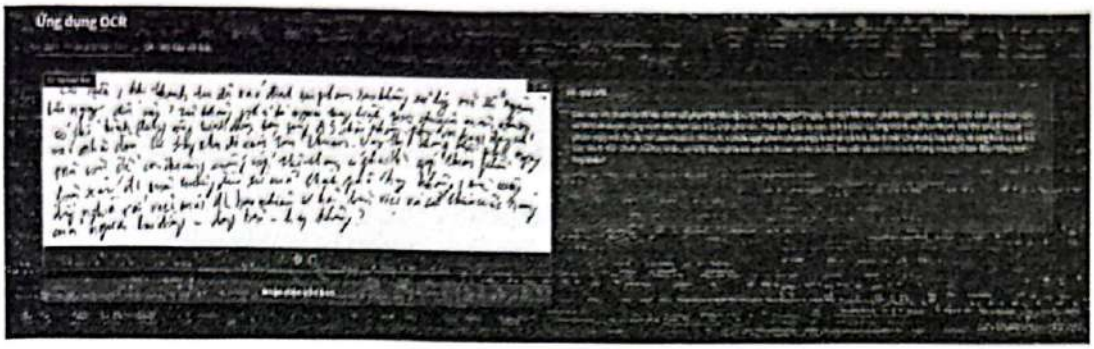
Một số thử nghiệm nhận diện với chữ in và chữ viết tay:



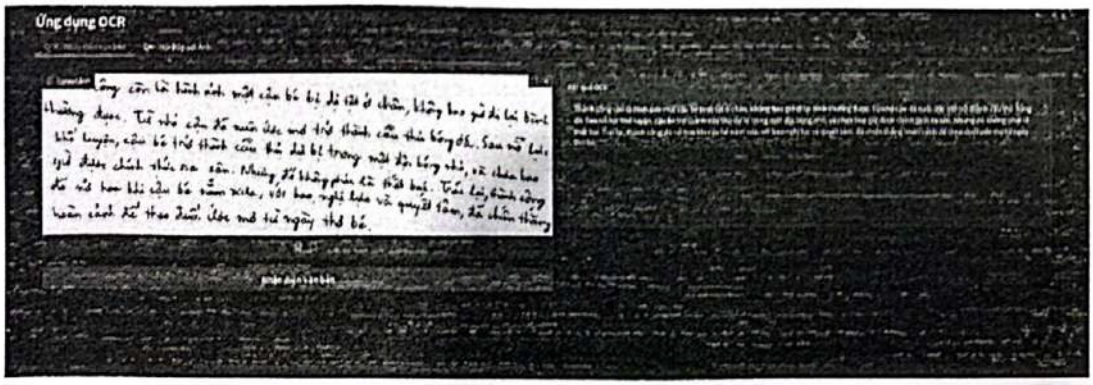
Hình 3.9: Thử nghiệm với hình ảnh đa ngôn ngữ



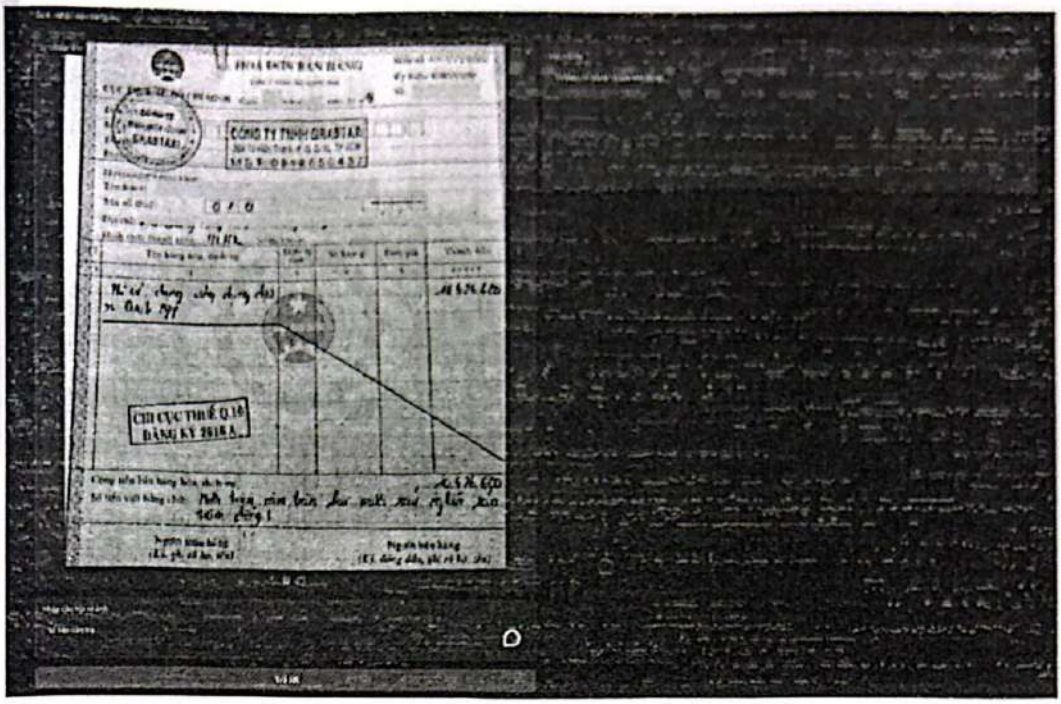
Hình 3.10: Thử nghiệm với chữ in



Hình 3.11: Thử nghiệm với chữ viết tay khó



Hình 3.12: Thử nghiệm với chữ viết tay



Hình 3.13: Thử nghiệm hỏi đáp thông tin

Nhận xét:

- Ứng dụng có khả năng xử lý chữ in đa ngôn ngữ, trích xuất hình ảnh chứa chữ in chính xác.
- Ứng dụng vẫn gặp hạn chế khi đối mặt với chữ viết tay quá khó, nét chữ quá liền nhau.
- Mô hình có khả năng trích xuất thông tin, ứng dụng trong các bài toán cần trích xuất các trường thông tin cụ thể.

3.7. Kết luận chương

Chương này đã chứng minh rằng việc tinh chỉnh mô hình Qwen 2.5-VL 7B bằng LoRA trên bộ dữ liệu tiếng Việt giúp mô hình đạt chất lượng OCR vượt trội, đặc biệt trên tài liệu khó và chữ viết tay. Các chỉ số đều giảm mạnh, mô hình hội tụ tốt và ổn định, đồng thời vượt xa các baseline truyền thống như PaddleOCR. Kết quả này khẳng định hướng tiếp cận dùng VLM cho OCR tiếng Việt là hoàn toàn khả thi và có tiềm năng ứng dụng rộng rãi trong các hệ thống số hóa tài liệu.

KẾT LUẬN

Đề án đã nghiên cứu và triển khai một hướng tiếp cận mới cho bài toán nhận dạng và trích xuất thông tin từ ảnh tài liệu tiếng Việt dựa trên mô hình thị giác–ngôn ngữ lớn Qwen 2.5-VL. Bằng cách tinh chỉnh mô hình thông qua kỹ thuật LoRA trên các bộ dữ liệu chữ in và chữ viết tay được chuẩn hóa, mô hình đã thể hiện khả năng học đặc thù tiếng Việt một cách hiệu quả. Các kết quả thực nghiệm cho thấy mô hình đạt độ chính xác vượt trội so với các baseline truyền thống như PaddleOCR và Vintern, đặc biệt ở các trường hợp văn bản khó, nhiễu hoặc viết tay.

Ngoài ra, mô hình sau khi tinh chỉnh còn cho thấy tính ứng dụng cao thông qua hệ thống demo có khả năng nhận dạng văn bản và trích xuất thông tin trực tiếp từ ảnh tài liệu. Điều này mở ra tiềm năng lớn cho việc triển khai thực tế trong các hệ thống số hóa dữ liệu, tự động hóa quy trình nhập liệu và các ứng dụng VQA dựa trên tài liệu tiếng Việt.

Mặc dù đạt được những kết quả khả quan, đề án vẫn còn một số hạn chế, như mô hình đôi khi chưa xử lý tốt các kiểu chữ viết tay quá phức tạp. Do vậy, các hướng phát triển tiếp theo có thể bao gồm mở rộng quy mô và đa dạng dữ liệu huấn luyện, kết hợp thêm các kỹ thuật tăng cường dữ liệu, và thử nghiệm các phiên bản VLM lớn hơn để cải thiện độ chính xác trên dữ liệu thực tế.

Nhìn chung, đề án đã chứng minh tính khả thi và hiệu quả của việc ứng dụng mô hình thị giác–ngôn ngữ lớn cho OCR tiếng Việt, đồng thời cung cấp một hướng tiếp cận hiện đại, tiềm năng cho các bài toán phân tích và xử lý tài liệu trong bối cảnh chuyển đổi số hiện nay.

TÀI LIỆU THAM KHẢO

1. Le, A., Lam, T., & Nguyen, D. (2025). "A Survey on Vietnamese Document Analysis and Recognition: Challenges and Future Directions". *arXiv preprint*.
2. Phan, T. H., Tran, D. C., & Hassan, M. F. (2021). "Vietnamese character recognition based on CNN model with reduced character classes". *Bulletin of Electrical Engineering and Informatics*, 10(2), 962–969.
3. Nguyen, Q. D. (2019). "OCR Error Correction for Unconstrained Vietnamese Handwritten Text". *Khoa học nghiên cứu*.
4. Công ty Cổ phần Kalapa. (2023). "Kalapa Challenge 2023 — Nhận diện chữ Tiếng Việt viết tay". *Báo Công An Nhân Dân – An ninh Thế giới*, 29/12/2023. Truy cập ngày 22/11/2025.
5. Nguyen, N. H., Vo, D. T. D., & Nguyen, K. V. (2022). "UIT-HWDB: Using Transferring Method to Construct A Novel Benchmark for Evaluating Unconstrained Handwriting Image Recognition in Vietnamese". *arXiv preprint*.
6. Smith, R. (2007). "An Overview of the Tesseract OCR Engine". In *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR)* (Vol. 2, pp. 629–633).
7. Liao, M., Wan, Z., Yao, C., Chen, K., & Bai, X. (2020). "Real-time Scene Text Detection with Differentiable Binarization". *AAAI Conference on Artificial Intelligence*, 34(7), 11474–11481.
8. Du, Y., Li, Z., Liu, W., Yin, F., & Zhou, M. (2020). "PaddleOCR: An Ultra Light-weight OCR System". *Baidu Research*, 8/2020.
9. Li, H., Wang, X., & Shen, C. (2019). "Towards End-to-End Text Spotting with Convolutional Recurrent Neural Networks". *Pattern Recognition*, 87, 12–27.
10. Shi, B., Bai, X., & Yao, C. (2016). "An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to

Scene Text Recognition". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11), 2298–2304.

11. Lee, D., Kim, J., Kim, D., Park, S., & Park, S. (2020). "On Recognizing Texts of Arbitrary Shapes with 2D Self-Attention". In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4560–4569).
12. Borisyuk, F., & Gordo, A. (2018). "Rosetta: Large Scale System for Text Detection and Recognition in Images". *KDD 2018*, 71–79.
13. Zhu, Y., Zheng, Z., Sun, Q., & Wu, H. (2022). "TrBA: Transformer-Based Bidirectional Attention for Scene Text Recognition". *arXiv preprint*.
14. Phạm, B. C. Q. (2020). "VietOCR – Transformer-based OCR for Vietnamese". *GitHub repository*. Truy cập tháng 7/2025.
15. Pham, H. Q., Nguyen, T. K.-B., Nguyen, Q. V., Tran, D. Q., Nguyen, N. H., Nguyen, K. V., & Luu-Thuy, N. (2024). "ViTextVQA: A Benchmark for Vietnamese Text Understanding in Visual Question Answering". *arXiv preprint*.
16. Nguyen, D. T., & Tran, A. K. (2023). "VinText: A Vietnamese Scene Text Dataset for Detection and Recognition". *Tạp chí Khoa học và Công nghệ Quân sự*, 63(5), 112–123.
17. Bai, Y., Xu, C., & Luo, Y. (2024). "Qwen-VL: A Versatile Vision-Language Model for Multilingual Document Understanding". *arXiv preprint*.
18. Hu, J., Shen, L., & Sun, G. (2018). "Squeeze-and-Excitation Networks". *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
19. Dosovitskiy, A., et al. (2020). "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". *ICLR*.
20. Radford, A., et al. (2021). "Learning Transferable Visual Models From Natural Language Supervision (CLIP)". *ICML*.
21. Li, J., et al. (2023). "LLaVA: Large Language and Vision Assistant". *arXiv preprint*.

22. Hu, E., et al. (2022). "LoRA: Low-Rank Adaptation of Large Language Models". *ICLR*.
23. Dettmers, T., et al. (2023). "QLoRA: Efficient Finetuning of Quantized Large Language Models". *NeurIPS*.
24. Zhang, J., et al. (2022). "Donut: Document Understanding Transformer Without OCR". *ECCV*.
25. Lee, H., et al. (2022). "Pix2Struct: Screenshot Parsing as Image-to-Text Generation". *arXiv preprint*.
26. Raffel, C., et al. (2020). "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5)". *JMLR*.
27. Nguyen, T. C., et al. (2024). "Vintern: A Lightweight Vietnamese Vision-Language Model for Document Understanding". *arXiv preprint*.
28. Matusu, S. (2020). "Character Error Rate and Word Error Rate for OCR Evaluation". *Computational Linguistics Notes*.
29. Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). "wav2vec 2.0: A Framework for Self-Supervised Speech Recognition". *NeurIPS*.
30. Tom Huynh. (2021). "Vietnamese Handwriting Recognition OCR Dataset". *GitHub repository*. https://github.com/TomHuynhSG/Vietnamese-Handwriting-Recognition

BẢN CAM ĐOAN

Tôi cam đoan đã thực hiện việc kiểm tra mức độ tương đồng nội dung đề án/luận văn/luận án qua phần mềm Turnitin một cách trung thực và đạt kết quả mức độ tương đồng ..19.. % toàn bộ nội dung đề án/luận văn/luận án. Bản đề án/luận văn/luận án kiểm tra qua phần mềm là bản cứng đề án/luận văn/luận án đã nộp để bảo vệ trước hội đồng. Nếu sai tôi xin chịu các hình thức kỷ luật theo quy định hiện hành của Học viện.

Hà Nội., ngày ..13.. tháng ..01.. năm 2026

HỌC VIÊN CAO HỌC/NCS

(Ký và ghi rõ họ tên)

Nguyễn Phương Anh

19% Tính tương đồng nói chung

Tổng cộng của tất cả các kết quả trùng khớp, bao gồm cả các nguồn trùng lặp, cho mỗi c...

Đã lọc khỏi Báo cáo

- Mục lục tham khảo
- Văn bản được trích dẫn
- Văn bản được trích dẫn
- Kết quả trùng khớp nhỏ (ít hơn 8 từ)

Nguồn hàng đầu

16% Nguồn Internet
8% Ấn bản
8% Bài tập được nộp (bài của học sinh)

Học viên

Nguyễn Phương Anh

Người hướng dẫn khoa học

TS. Nguyễn Đình Hóa