

ĐẠI HỌC QUỐC GIA TP.HCM  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO BÀI TẬP THỰC HÀNH 2  
Nhập môn Xử lý ngôn ngữ tự nhiên -  
23CNTThuc2

Fine-tune Deepseek-OCR with Vietnamese  
Dataset

Giảng viên hướng dẫn: Dinh Diền  
Nguyễn Hồng Bửu Long  
Lương An Vinh

Sinh viên thực hiện: 23127485 – Phạm Quang Thịnh

## Mục lục

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Giới thiệu</b>                                   | <b>1</b> |
| <b>2</b> | <b>Phương pháp luận</b>                             | <b>1</b> |
| 2.1      | Chuẩn bị dữ liệu (Data Preparation) . . . . .       | 1        |
| 2.2      | Kỹ thuật Huấn luyện . . . . .                       | 2        |
| 2.3      | Cấu hình Thực nghiệm . . . . .                      | 2        |
| <b>3</b> | <b>Thực nghiệm và Đánh giá</b>                      | <b>2</b> |
| 3.1      | Thiết lập thí nghiệm . . . . .                      | 2        |
| 3.2      | Độ đo đánh giá (Evaluation Metrics) . . . . .       | 3        |
| 3.3      | Kịch bản đánh giá . . . . .                         | 3        |
| <b>4</b> | <b>Kết quả và Thảo luận</b>                         | <b>3</b> |
| 4.1      | Kết quả định lượng (Quantitative Results) . . . . . | 3        |
| 4.2      | Phân tích quá trình huấn luyện . . . . .            | 4        |
| 4.3      | Kết quả trực quan . . . . .                         | 4        |
| 4.4      | Thảo luận và Hạn chế . . . . .                      | 5        |
| <b>5</b> | <b>Kết luận</b>                                     | <b>5</b> |
|          | <b>Mã nguồn và Tài nguyên</b>                       | <b>6</b> |
|          | <b>Tài liệu tham khảo</b>                           | <b>6</b> |

## 1. Giới thiệu

Trong lĩnh vực Xử lý ngôn ngữ tự nhiên, bài toán Nhận dạng ký tự quang học (OCR) cho tiếng Việt luôn đặt ra nhiều thách thức đặc thù so với các ngôn ngữ Latin khác. Mặc dù các Mô hình Ngôn ngữ Đa phương thức Lớn (LMMs) hiện đại đã đạt được nhiều thành tựu, việc ứng dụng trực tiếp chúng vào văn bản tiếng Việt vẫn gặp hai trở ngại lớn:

- Độ phức tạp của hệ thống chữ viết:** Tiếng Việt sở hữu hệ thống dấu thanh phong phú (sắc, huyền, hỏi, ngã, nặng) và các nguyên âm có dấu (ă, â, ê, ô, ơ, ư). Sự chồng chéo của các dấu này, đặc biệt trong **văn bản viết tay hoặc ảnh chất lượng thấp**, dễ gây nhầm lẫn cho máy tính.
- Hạn chế của mô hình nền tảng:** Các mô hình LMM phổ biến hiện nay (như Llama-Vision, Qwen-VL...) thường được huấn luyện chủ yếu trên tập dữ liệu tiếng Anh hoặc tiếng Trung. Do đó, chúng thiếu "tri thức" sâu về hình thái ngữ pháp tiếng Việt, dẫn đến tỷ lệ lỗi ký tự (CER) cao khi phải xử lý các văn bản tiếng Việt thuần túy mà chưa qua tinh chỉnh.

Dồ án này tập trung giải quyết vấn đề trên bằng phương pháp tinh chỉnh (Fine-tuning) mô hình **DeepSeek-OCR** sử dụng bộ dữ liệu tiếng Việt chuyên biệt. Áp dụng kỹ thuật **QLoRA** để tối ưu hóa quá trình huấn luyện trên tài nguyên hạn chế, với mục tiêu chính là giảm thiểu sai sót nhận dạng và nâng cao độ chính xác tổng thể của mô hình.

## 2. Phương pháp luận

### 2.1. Chuẩn bị dữ liệu (Data Preparation)

Dữ liệu đóng vai trò cốt lõi trong việc tinh chỉnh mô hình. Quy trình xử lý dữ liệu được thực hiện qua các bước sau:

- Thu thập:** Sử dụng bộ dữ liệu ảnh chữ viết tay tiếng Việt kèm nhãn (ground truth) tương ứng (bộ dữ liệu **UIT\_HWDB\_line** được gợi ý trong đề).
- Tiền xử lý văn bản:** Chuẩn hóa toàn bộ nhãn văn bản về định dạng Unicode NFC, chuyển về chữ thường (lowercase) và loại bỏ các khoảng trắng thừa để đảm bảo tính nhất quán khi tính toán hàm mất mát (Loss). Sử dụng **DeepSeekOCRDataCollator** để chuẩn hóa kích thước ảnh (base size 1024, patch size 640) và tạo mask cho văn bản tiếng Việt.
- Định dạng đầu vào:** Dữ liệu được chuyển đổi sang định dạng JSONL (Conversation format) phù hợp với các mô hình VLM:
  - Input:* Ảnh + Prompt hướng dẫn (“<image>\Free OCR.”).
  - Output:* Văn bản tiếng Việt trong ảnh.
- Phân chia dữ liệu (Data Splitting):** Để đảm bảo tính khách quan và kiểm soát hiện tượng overfitting, tập dữ liệu được phân chia thành 3 phần riêng biệt:
  - Train set:* **3.156 mẫu** - Được sử dụng chính để cập nhật trọng số của mô hình.
  - Validation set:* **150 mẫu** - Được sử dụng để đánh giá định kỳ trong quá trình huấn luyện nhằm theo dõi hàm mất mát (Loss) và lựa chọn checkpoint tối ưu.
  - Test set:* **400 mẫu** - Được giữ độc lập hoàn toàn (unseen data), không tham gia vào quá trình học, dùng để tính toán chỉ số CER/WER cuối cùng cho báo cáo.

## 2.2. Kỹ thuật Huấn luyện

Để giải quyết bài toán huấn luyện mô hình lớn trên tài nguyên hạn chế (Google Colab T4 - 16GB VRAM), thư viện **Unsloth** kết hợp với kỹ thuật **QLoRA** (Quantized Low-Rank Adaptation) được sử dụng cho bài toán này.

Cụ thể:

- **4-bit Quantization:** Mô hình gốc được nén (load) ở độ chính xác 4-bit, giúp giảm dung lượng VRAM tiêu thụ từ 15GB xuống còn 6-7GB mà vẫn giữ được hiệu năng gần tương đương 16-bit, cho phép chạy được trên GPU T4.
- **LoRA Adapters:** Thay vì cập nhật toàn bộ tham số của mô hình (Full Fine-tuning), ta đóng băng (freeze) mô hình gốc và chỉ huấn luyện các ma trận trọng số nhỏ (Adapters) được gắn thêm vào các lớp Attention (q\_proj, k\_proj, v\_proj, o\_proj).

## 2.3. Cấu hình Thực nghiệm

Các tham số siêu hình (Hyperparameters) được thiết lập như Bảng 1 dưới đây nhằm đảm bảo sự ổn định và tối ưu hóa hàm mất mát:

Bảng 1: Các tham số huấn luyện chi tiết

| Tham số                        | Giá trị thiết lập                    |
|--------------------------------|--------------------------------------|
| Framework                      | Unsloth (PyTorch + Transformers)     |
| Phương pháp tối ưu (Optimizer) | AdamW 8-bit                          |
| Learning Rate                  | $2 \times 10^{-4}$ (Linear Schedule) |
| Batch Size (per device)        | 2                                    |
| Gradient Accumulation Steps    | 4 (Tương đương Batch size tổng = 8)  |
| Số bước huấn luyện (Max Steps) | 150                                  |
| Độ chính xác (Precision)       | FP16 (Mixed Precision)               |
| Evaluation Strategy            | Steps (Mỗi 10 bước đánh giá 1 lần)   |

## 3. Thực nghiệm và Đánh giá

### 3.1. Thiết lập thí nghiệm

Quy trình thực nghiệm được thiết kế nhằm so sánh công bằng hiệu năng giữa mô hình gốc (Baseline) chưa qua chỉnh sửa và mô hình sau khi tinh chỉnh (Fine-tuned). Các thí nghiệm được thực hiện trên cùng một tập dữ liệu kiểm tra (Test set) gồm **400 mẫu** độc lập.

Môi trường thực nghiệm được triển khai trên nền tảng Google Colab với cấu hình phần cứng:

- **GPU:** NVIDIA Tesla T4 (16GB VRAM).
- **Thư viện hỗ trợ:** Unsloth, PyTorch, Transformers.

Quá trình huấn luyện sử dụng hàm mất mát **Cross-Entropy Loss** tiêu chuẩn cho các mô hình sinh ngôn ngữ, với mục tiêu tối thiểu hóa sai lệch giữa phân phối xác suất của từ dự đoán và từ thực tế (Ground truth).

### 3.2. Độ đo đánh giá (Evaluation Metrics)

Để đánh giá chất lượng hệ thống OCR, ta sử dụng hai độ đo phổ biến nhất trong cộng đồng nghiên cứu:

1. **Character Error Rate (CER):** Được tính dựa trên khoảng cách Levenshtein (số thao tác thêm, sửa, xóa ký tự tối thiểu để biến chuỗi dự đoán thành chuỗi đích).

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

Trong đó:  $S$  là số ký tự thay thế,  $D$  là số ký tự xóa,  $I$  là số ký tự thêm vào, và  $N$  là tổng số ký tự của nhãn gốc. *Lý do sử dụng:* Đối với tiếng Việt, CER là độ đo quan trọng nhất vì nó phản ánh chính xác khả năng nhận diện các dấu thanh nhỏ (ví dụ: phân biệt 'a', 'á', 'à').

2. **Word Error Rate (WER):** Tương tự như CER nhưng tính trên cấp độ từ (word-level), đánh giá khả năng hiểu và tái tạo ngữ nghĩa của văn bản.

### 3.3. Kịch bản đánh giá

Tiến hành hai kịch bản đánh giá:

- **Kịch bản 1 (Baseline Zero-shot):** Sử dụng mô hình gốc, cung cấp ảnh và prompt “Free OCR”, yêu cầu mô hình sinh văn bản mà không qua bất kỳ bước cập nhật trọng số nào. Kết quả này đóng vai trò làm mốc so sánh cơ sở.
- **Kịch bản 2 (Fine-tuned Inference):** Sử dụng mô hình sau 150 bước huấn luyện, load lại trọng số LoRA adapters và thực hiện suy diễn trên cùng tập dữ liệu.

## 4. Kết quả và Thảo luận

### 4.1. Kết quả định lượng (Quantitative Results)

Bảng 2 trình bày hiệu năng của mô hình trước và sau khi tinh chỉnh trên tập dữ liệu kiểm tra (Test set).

Bảng 2: So sánh độ lỗi giữa Baseline và Fine-tuned Model

| Mô hình    | CER (%) | WER (%) | Cải thiện CER | Cải thiện WER |
|------------|---------|---------|---------------|---------------|
| Baseline   | 31.92%  | 66.69%  | -             | -             |
| Fine-tuned | 14.05%  | 33.31%  | +17.87%       | 33.39%        |

**Nhận xét:** Sau 150 bước huấn luyện, mô hình đã đạt được mức giảm lỗi ký tự (CER) khá quan trọng xuống còn 14%. Kết quả này khẳng định tính hiệu quả của phương pháp QLoRA trong việc thích nghi mô hình ngôn ngữ lớn với ngôn ngữ tiếng Việt.

## 4.2. Phân tích quá trình huấn luyện

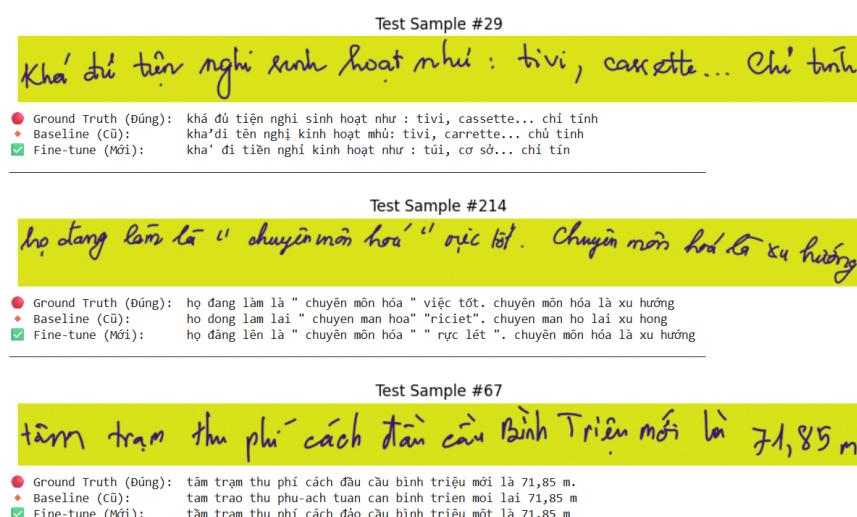
Sự ổn định và hiệu quả của quá trình huấn luyện được thể hiện rõ nét qua sự hội tụ của hàm mất mát trên tập Validation (Validation Loss). Cụ thể, Loss giảm liên tục và mượt mà từ **0.860** xuống mức rất thấp là **0.553** sau 150 bước. Mặc dù Training Loss có sự dao động do kích thước batch nhỏ, nhưng việc Validation Loss giảm sâu và chưa có dấu hiệu đi ngang cho thấy mô hình đã tổng quát hóa tốt các đặc trưng dữ liệu và hoàn toàn không xuất hiện hiện tượng quá khớp (overfitting).

| [150/150 2:05:11, Epoch 0/1] |               |                 |
|------------------------------|---------------|-----------------|
| Step                         | Training Loss | Validation Loss |
| 10                           | 0.825400      | 0.860202        |
| 20                           | 0.991800      | 0.782285        |
| 30                           | 0.901700      | 0.733513        |
| 40                           | 0.931000      | 0.711538        |
| 50                           | 0.750900      | 0.668216        |
| 60                           | 0.413000      | 0.634570        |
| 70                           | 0.760700      | 0.627119        |
| 80                           | 0.642100      | 0.616765        |
| 90                           | 0.576300      | 0.605120        |
| 100                          | 0.596100      | 0.591981        |
| 110                          | 0.649900      | 0.580551        |
| 120                          | 0.621000      | 0.573014        |
| 130                          | 0.674800      | 0.561192        |
| 140                          | 0.759200      | 0.555314        |
| 150                          | 0.632000      | 0.552983        |

Hình 1: Bảng giá trị Validation Loss theo thời gian

## 4.3. Kết quả trực quan

Hình 2 minh họa sự khác biệt rõ rệt trong đầu ra của hai mô hình:



Hình 2: So sánh thực tế: Baseline và Fine-tuned

### Phân tích chi tiết từ hình ảnh:

- **Khả năng phục hồi dấu thanh:** Đây là cải thiện rõ rệt nhất. Quan sát mẫu #67 và #214, mô hình Baseline gần như bỏ qua toàn bộ dấu tiếng Việt hoặc đoán sai hoàn toàn (ví dụ: “*bình triệu*” bị biến thành “*binh trien*”, “*xu hướng*” thành “*xu hong*”). Ngược lại, mô hình Fine-tuned nhận diện và đặt dấu chính xác cho các cụm từ phức tạp như “*chuyên môn hóa*”, “*thu phí*”.
- **Khắc phục từ vô nghĩa:** Khi gặp các nét chữ cầu thả hoặc dính nét, Baseline thường sinh ra các chuỗi ký tự không có trong từ điển (ví dụ: “*riciet*” ở mẫu #214 hay “*mhủ*” ở mẫu #29). Mô hình Fine-tuned, nhờ được huấn luyện lại, đã học được cấu trúc từ vựng tiếng Việt nên luôn cố gắng sinh ra các từ có nghĩa, giúp câu văn trở nên dễ hiểu hơn ngay cả khi ảnh đầu vào có chất lượng thấp.

#### 4.4. Thảo luận và Hạn chế

Mặc dù kết quả khả quan, hệ thống vẫn còn một số thách thức cần giải quyết:

- **Giới hạn phần cứng:** Việc sử dụng GPU T4 buộc phải giới hạn Batch size nhỏ (2). Nếu được huấn luyện trên phần cứng mạnh hơn (A100) với độ dài ngữ cảnh lớn hơn, mô hình có thể xử lý tốt hơn các văn bản dài.
- **Dữ liệu huấn luyện:** Số lượng 3.156 mẫu tuy đủ để hội tụ nhưng vẫn khiêm tốn so với sự đa dạng của chữ viết tay. Việc mở rộng tập dữ liệu hứa hẹn sẽ giảm CER xuống dưới 10%.

### 5. Kết luận

Đồ án này đã xây dựng và thực nghiệm thành công quy trình tinh chỉnh (Fine-tuning) Mô hình Ngôn ngữ Đa phương thức Lớn cho bài toán nhận dạng văn bản tiếng Việt. Việc áp dụng kỹ thuật **QLoRA** thông qua thư viện Unslloth đã chứng minh được tính khả thi của việc huấn luyện các mô hình AI hiện đại trên tài nguyên phần cứng hạn chế (GPU T4) mà vẫn đảm bảo hiệu năng cao.

Các đóng góp và kết quả chính của đồ án bao gồm:

- **Cải thiện độ chính xác:** Mô hình sau khi tinh chỉnh đạt tỷ lệ lỗi ký tự (CER) **14%** và tỷ lệ lỗi từ (WER) **33.31%**. Đây là mức cải thiện vượt bậc so với mô hình Baseline, đặc biệt trong việc xử lý chính xác hệ thống dấu thanh và nguyên âm phức tạp của tiếng Việt.
- **Khả năng xử lý dữ liệu thực tế:** Thay vì chỉ nhận dạng tốt các văn bản in chuẩn, mô hình đã học được cách xử lý các biến thể đa dạng của **chữ viết tay** và các ảnh có chất lượng không đồng đều. Điều này khắc phục hạn chế lớn nhất của các mô hình nền tảng vốn chưa được tối ưu cho đặc thù chữ viết người Việt.
- **Tối ưu hóa tài nguyên:** Đồ án khẳng định rằng với phương pháp tiếp cận đúng đắn (sử dụng Quantization và Low-Rank Adaptation), sinh viên và cộng đồng nghiên cứu có thể tự xây dựng các mô hình OCR chuyên biệt chất lượng cao mà không nhất thiết phải phụ thuộc vào các siêu máy tính đắt tiền.

Kết quả của đồ án đóng vai trò là tiền đề vững chắc cho việc ứng dụng các mô hình ngôn ngữ thị giác (VLM) vào các bài toán OCR tài liệu tiếng Việt trong tương lai.

## Mã nguồn và Tài nguyên (Source Code)

Toàn bộ mã nguồn huấn luyện, đánh giá và các tài liệu liên quan của đồ án được lưu trữ công khai tại GitHub Repository dưới đây:

- **GitHub Repository:**

[https://github.com/pqthinh232/Fine-tune-Deepseek-OCR-with-Vietnamese-Dataset.  
git](https://github.com/pqthinh232/Fine-tune-Deepseek-OCR-with-Vietnamese-Dataset.git)

- **Nội dung bao gồm:**

- Notebook huấn luyện và đánh giá (.ipynb) với đầy đủ log chạy thực tế.
- Hướng dẫn cài đặt môi trường và tái lập kết quả.
- Link tải Model Checkpoint (đã lưu trên Google Drive).

## Tài liệu

- [1] Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). *QLoRA: Efficient Finetuning of Quantized LLMs*. arXiv preprint arXiv:2305.14314. <https://arxiv.org/pdf/2305.14314.pdf>
- [2] Unsloot AI. *DeepSeek-OCR: How to Run & Fine-tune*. Truy cập ngày 15/01/2025. <https://docs.unsloot.ai/models/deepseek-ocr-how-to-run-and-fine-tune>
- [3] 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”. *2022 RIVF International Conference on Computing and Communication Technologies*. IEEE, pp. 659–664. <https://github.com/nghiangh/UIT-HWDB-dataset>