ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

BÁO CÁO TIẾN ĐỘ

Tối ưu hóa và Triển khai mô hình ngôn ngữ lớn (LLM) trên nền tảng CPU cho tác vụ hỗ trợ lập trình nội bộ (On-Device Code Completion).

Người hướng dẫn: KS. Lê Ngọc Thiện

Người thực hiện: Mạc Phạm Thiên Long

Hà Nội, ngày 19 tháng 12 năm 2025

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

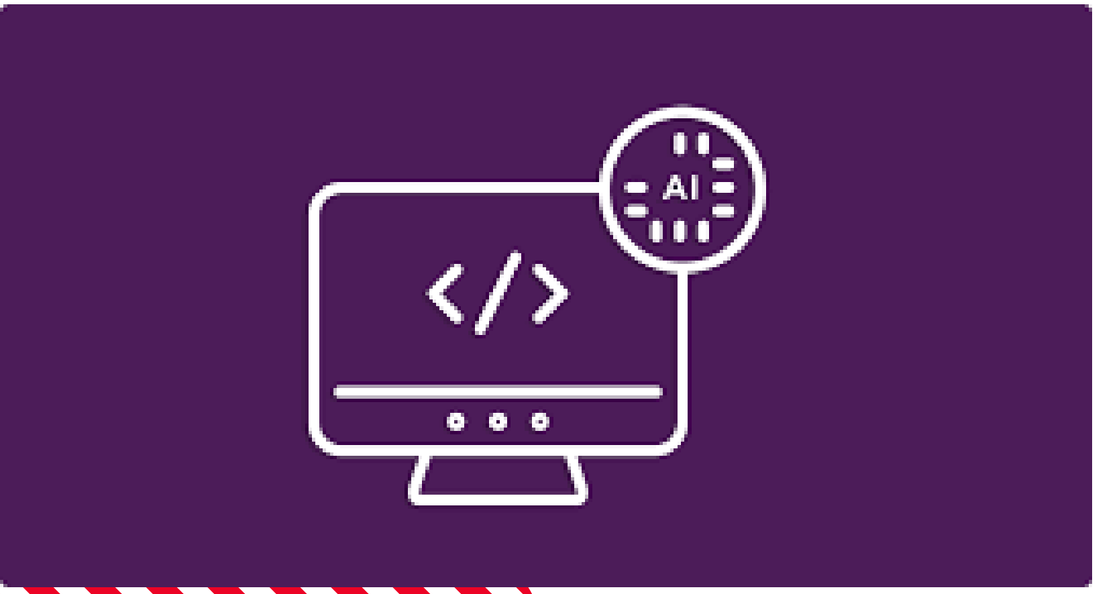
viettel networks

MỤC LỤC

Bối cảnh

- Việc sử dụng các Cloud LLM (GitHub Copilot, ChatGPT) tiềm ẩn rủi ro rò rỉ mã nguồn nội bộ và các thông tin nhạy cảm.
- Triển khai LLM truyền thống đòi hỏi hệ thống GPU (Nvidia A100/H100) đắt đỏ, khó phổ cập cho mọi lập trình viên trên máy tính cá nhân.
- Kết nối cloud thường có độ trễ cao, gây mất nhịp tập trung khi coding.

Mục tiêu đề tài



Xây dựng hệ thống AI Coding Assistant chuyên biệt với 3 trụ cột chính

- Hoạt động hoàn toàn Offline, dữ liệu không rời khỏi máy cá nhân
- Chạy mượt mà trên CPU desktop/laptop thông qua các kỹ thuật nén mô hình tiên tiến.
- Được tinh chỉnh (Fine-tuning) để hiểu sâu các ngôn ngữ lập trình phổ biến (Python, Java, C++) và ngữ cảnh riêng của từng dự án.

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Nhiệm vụ trọng tâm

- Dữ liệu: Làm sạch codebase, áp dụng kỹ thuật Import Dropout và Indentation Tokens để tăng khả năng hiểu cấu trúc code.
- Huấn luyện: Sử dụng kỹ thuật FIM (Fill-In-the-Middle) kết hợp LoRA/QLoRA để tinh chỉnh mô hình 0.5B tham số.
- Nén mô hình: Lượng tử hóa 4-bit (GGUF) giúp giảm dung lượng RAM nhưng vẫn giữ 95%+ độ chính xác của mô hình gốc.

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Các chỉ số đánh giá

Offline	***Online***
Edit similarity (ES), Exact match (EM), Matched ratio (MR), Perfect lines (PL).	*Acceptance Rate, Time-to-Start, và Ratio-of-Completed Code (RoCC0*

Các chỉ số đánh giá: Offline

Tập trung vào tính chính xác về mặt nội dung của mô hình so với đoạn code gốc

1

Edit Similarity (ES)

Đo lường mức độ tương tự giữa code gợi ý và code thực tế (dựa trên khoảng cách Levenshtein)

2

Exact Match (EM)

Tỉ lệ khớp chính xác 100% từng ký tự. Phản ánh khả năng dự đoán hoàn hảo của mô hình.

3

Perfect Lines (PL)

Tỉ lệ số dòng code khớp hoàn toàn trên tổng số dòng mong muốn. Rất quan trọng trong tác vụ sinh mã nhiều dòng.

4

Matched Ratio (MR) và RoCC (Ratio of Completed Code

Đánh giá tỉ lệ các đoạn mã và ký tự xuất hiện đúng vị trí, giúp hiểu rõ hơn về phân bổ lỗi của mô hình.

Các chỉ số đánh giá: Online

Tập trung vào khả năng vận hành thực tế trên CPU máy tính văn phòng/laptop.

1

Time To Start (TTS) / Latency

Thời gian từ lúc người dùng dừng gõ đến khi gợi ý đầu tiên xuất hiện.
Mục tiêu cụ thể: > 200ms

2

Tokens Per Second (TPS)

Tốc độ sinh mã trung bình. Đảm bảo gợi ý xuất hiện tức thì ngay cả với các đoạn mã dài.

3

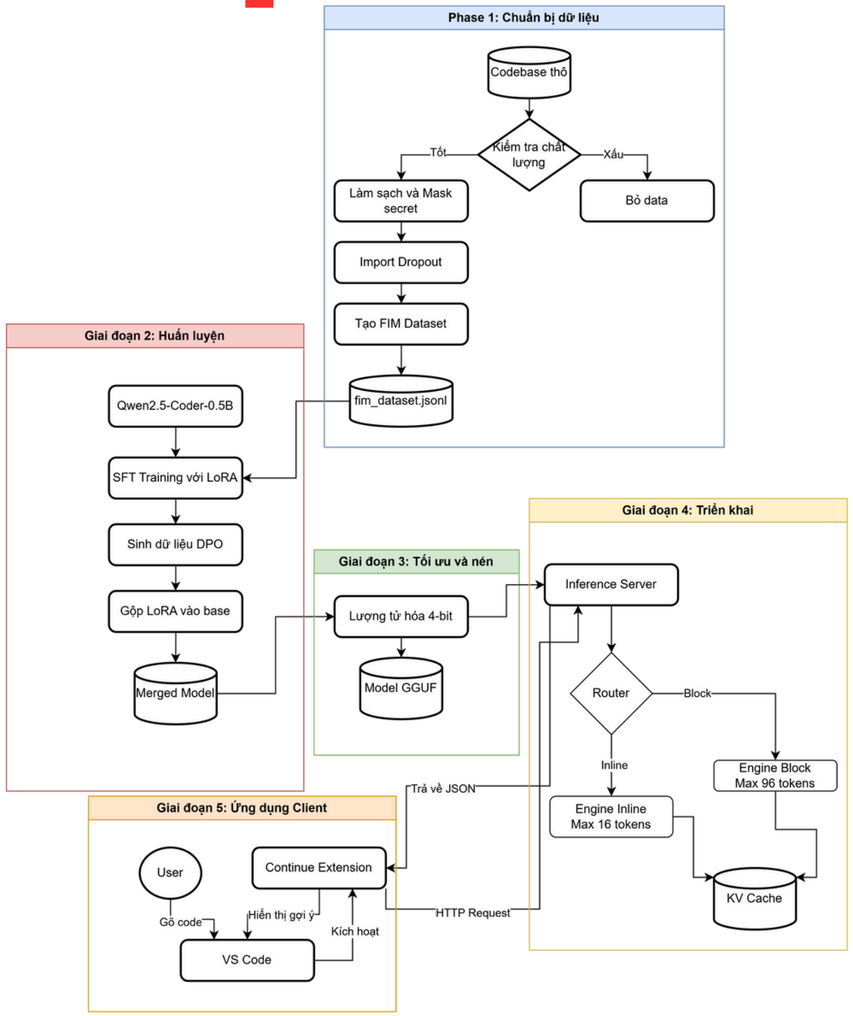
Acceptance Rate (AR)

Chỉ số mô phỏng mức độ hữu ích của mô hình (được tính bằng tỉ lệ code gợi ý vượt qua ngưỡng ES nhất định).

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Pipeline



ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Giai đoạn 1: Chuẩn bị dữ liệu

1

Thu thập dữ liệu

- Dữ liệu lấy từ bộ dữ liệu The Stack (xấp xỉ 300k dữ liệu)

Lọc theo các tiêu chí:

- File size: 1 - 100KB
- Loại bỏ file auto-generated, minified, binary
- Chỉ giữ extension: .py, .java, .cpp, .c, .h

2

Làm sạch dữ liệu

- Xóa comments: Loại bỏ docstrings, inline comments
- Mask secrets: Thay API keys, passwords bằng <SECRET>
- Normalize whitespace: Chuẩn hóa indentation về 4 spaces
- Loại bỏ file trùng lặp

3

Tăng cường dữ liệu

- Import dropout (30%): Giúp model học suy luận thay vì copy pattern
- Indentation tokens: Giúp model hiểu scope và cấu trúc lồng nhau

4

Tạo Dataset FIM (Fill-In-Middle)

Format chuẩn:
<PRE>prefix_code
<SUF>suffix_code
<MID>target_completion

5

Chia Dataset

- Train: 90%
- Validation: 5%
- Test: 5%

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Giai đoạn 2: Huấn luyện

Các tham số huấn luyện

01

Base Model: Qwen2.5-Coder-0.5B

Nhỏ gọn, tối ưu cho code

02

LoRA Rank: 64

Cân bằng giữa capacity và memory

03

LoRA Alpha: 128

Tỉ lệ chuẩn (2x rank)

04

Learning Rate: 2e-4

Phù hợp với LoRA

05

Batch Size: 4

Tối đa với VRAM của GPU

06

Gradient Accumulation: 8

Effective batch = 32
Chia batch để tăng tốc độ huấn luyện

07

Epochs: 3

Tránh overfitting

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Giai đoạn 3

SFT Train

- Mục tiêu: Giúp mô hình học cấu trúc code và cách hoàn thiện câu lệnh (Fill-In-the-Middle).
- Dataset:Từ hơn 300k data đã crawl và filter → chia dataset thành 3 phần: 241k train data, 14k validation data, 45k test data

DPO (Direct Preference Optimization)

- Mục tiêu: Tinh chỉnh sở thích (Alignment), giúp mô hình chọn cách viết code tối ưu hơn, tránh các pattern lỗi hoặc dư thừa.
- Dataset: Cặp dữ liệu so sánh (Chosen vs Rejected)
- Tạo ra hơn 1400 cặp Preferences Chosen - Rejected

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

DPO (Direct Preference Optimization)

Tại sao ta cần DPO?

- SFT đôi khi khiến mô hình bị học vẹt hoặc tạo ra code chạy được nhưng không tối ưu.
- DPO giúp mô hình phân biệt giữa code chạy được (Rejected) và code chuẩn (Chosen).

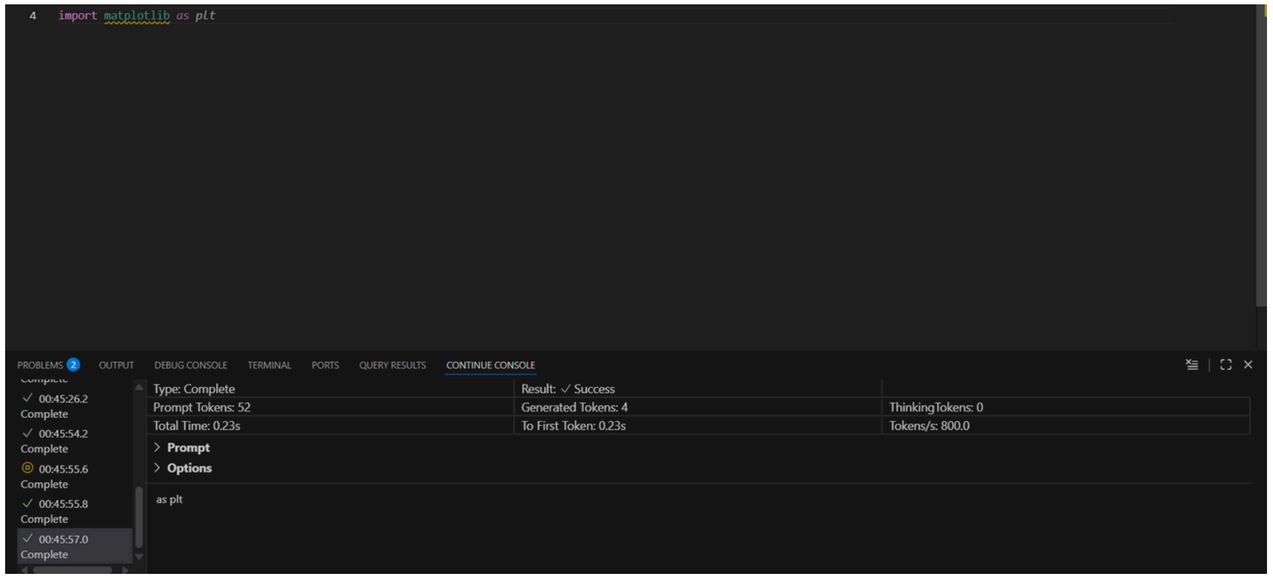
Cách thức thực hiện:

- Sử dụng cơ chế so sánh trực tiếp Log-Likelihood giữa hai đáp án.
- Không cần mô hình Reward rời rạc (tối ưu hơn so với RLHF truyền thống).

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Giai đoạn 4 + 5



- Kiến trúc triển khai:
  - Local Server: Xây dựng Inference Server bằng FastAPI và llama-cpp-python.
  - Dual Engine: Tách biệt luồng xử lý Inline (nhanh, ngắn) và Block (dài, chi tiết) để tối ưu trải nghiệm.
- Tích hợp IDE: Kết nối trực tiếp với VS Code thông qua Continue Extension.
  - Cơ chế Debounce (300ms) để tránh lãng phí tài nguyên CPU.
  - Hỗ trợ hiển thị dạng Ghost text (văn bản mờ) và nhập nhanh bằng phím Tab.
- Tính năng bổ sung: Tích hợp bộ đệm KV Cache để xử lý context code dài mà không làm tăng độ trễ.

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Kết quả

Mô hình	ES	EM	PL	MR	RoCC	TTS
Base Model(Qwen 2.5-Coder-0.5B instructed)	0.1%	7.42%	1.53%	8.12%	2.99%	174.04ms
SFT Merged Model	55.36%	22%	27.4%	61.24%	39.17%	173.49ms
DPO Merged Model	55.19%	21.66%	27.11%	61.07%	38.93%	178.055ms
SFT Quantized*	2.05%	1.22%	0.21%	0.89%	0.65%	668.14ms
DPO Quantized*	1.96%	1.22%	0.2%	0.82%	0.65%	672.089ms

*** Test thực tế trên laptop: EM 66.7%, ES 76.7%, Latency 170ms**

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Nhận xét

Đột phá từ quá trình huấn luyện SFT:

- Chỉ số EM tăng từ 0.1% lên 21.78% (gấp 217 lần so với Model gốc).
- Chỉ số ES đạt 55.33%, khẳng định mô hình đã làm chủ hoàn toàn định dạng FIM và hỗ trợ tốt logic lập trình thực tế.
- Các chỉ số MR (61.23%) và RoCC (38.96%) cho thấy khả năng sinh code phù hợp ngữ cảnh đạt mức tốt.

Về DPO và Mô hình đã Lượng tử hóa::

- DPO Merged giữ nguyên hiệu suất của SFT (21.66% EM).
- Bản Quantized (GGUF) đạt latency ~170ms trên CPU laptop, đáp ứng yêu cầu real-time.
- Độ chính xác GGUF trên test thực tế: EM 66.7% (inline), ES 76.7%.

Mô hình Quantized sẵn sàng triển khai on-device với latency thấp và độ chính xác chấp nhận được, phù hợp cho autocomplete code thời gian thực (Tuy nhiên cần đánh giá thêm)

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

Kết luận

Thành tựu chính

- Xây dựng thành công quy trình huấn luyện từ SFT đến DPO cho mô hình Code LLM.
- Nén mô hình xuống format 4-bit GGUF, cho phép chạy mượt mà trên CPU laptop
- 100% dữ liệu xử lý Offline, không gửi dữ liệu ra ngoài, đảm bảo an toàn

Hạn chế

- Kích thước 0.5B vẫn còn giới hạn khi gặp các cấu trúc logic lập trình quá phức tạp
- Hiện tại mới chỉ hỗ trợ ngữ cảnh trong một tệp duy nhất (single-file context), chưa đọc được toàn bộ codebase.
- Hệ thống chưa tích hợp cơ chế Retrieval để cá nhân hóa gợi ý
- DPO gặp tình trạng overfitting nên chưa thể deploy

Tầm nhìn tương lai

- Chuyển đổi sang model 1.5B - 3B kết hợp kỹ thuật Speculative Decoding để giữ nguyên tốc độ nhưng tăng chất lượng.
- Tích hợp RAG với Embedding model on-device để hiểu sâu hơn về cấu trúc dự án.
- Mở rộng dataset để hỗ trợ chuyên sâu cho các ngôn ngữ khác
- Cải tiến thêm về DPO
- Đánh giá lại các model quantized sau khi train lại DPO và điều chỉnh calculation metrics

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG QUỐC TẾ
VNU-INTERNATIONAL SCHOOL

viettel networks

THANK YOU!

Ngày 19 tháng 12 năm 2025