**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**ĐỒ ÁN TỐT NGHIỆP**

**Large Language Model**

Anh.PQ213808@sis.hust.edu.vn

**Ngành Điện tử - Viễn thông**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | TS. Trần Mạnh Hoàng  Chữ ký của GVHD |
| **KHOA:** | Điện tử |
| **HÀ NỘI, 6/2025**  **Đánh giá quyển đồ án tốt nghiệp**  **(Dùng cho giảng viên hướng dẫn)**  Giảng viên đánh giá: ..............................................................................................  Họ và tên Sinh viên:................................................ MSSV:...................................  Tên đồ án: ...............................................................................................................  .................................................................................................................................  ***Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:***  ***Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)***   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | | | | 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | | 1 | 2 | 3 | 4 | 5 | | 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | | 1 | 2 | 3 | 4 | 5 | | 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | | 1 | 2 | 3 | 4 | 5 | | 4 | Có kết quả mô phỏng/thưc nghiệm và trình bày rõ ràng kết quả đạt được | | 1 | 2 | 3 | 4 | 5 | | **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | | | | 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | | 1 | 2 | 3 | 4 | 5 | | 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng. | | 1 | 2 | 3 | 4 | 5 | | 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai. | | 1 | 2 | 3 | 4 | 5 | | **Kỹ năng viết (10)** | | | | | | | | | 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | | 1 | 2 | 3 | 4 | 5 | | 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | | 1 | 2 | 3 | 4 | 5 | | **Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)** | | | | | | | | | 10a | | Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế | 5 | | | | | | 10b | | Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest. | 2 | | | | | | 10c | | Không có thành tích về nghiên cứu khoa học | 0 | | | | | | **Điểm tổng** | | | **/50** | | | | | | **Điểm tổng quy đổi về thang 10** | | |  | | | | |   ***Nhận xét thêm của Thầy/Cô (****giảng viên hướng dẫn nhận xét về thái độ và tinh thần làm việc của sinh viên****)***  ................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................    Ngày: ... / ... / 20..  **Người nhận xét**  (Ký và ghi rõ họ tên)  **Đánh giá quyển đồ án tốt nghiệp**  **(Dùng cho cán bộ phản biện)**  Giảng viên đánh giá: ..............................................................................................  Họ và tên Sinh viên:................................................ MSSV:...................................  Tên đồ án: ...............................................................................................................  .................................................................................................................................  ***Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:***  ***Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)***   |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **Có sự kết hợp giữa lý thuyết và thực hành (20)** | | | | | | | | | 1 | Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án | | 1 | 2 | 3 | 4 | 5 | | 2 | Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế) | | 1 | 2 | 3 | 4 | 5 | | 3 | Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề | | 1 | 2 | 3 | 4 | 5 | | 4 | Có kết quả mô phỏng/thưc nghiệm và trình bày rõ ràng kết quả đạt được | | 1 | 2 | 3 | 4 | 5 | | **Có khả năng phân tích và đánh giá kết quả (15)** | | | | | | | | | 5 | Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống | | 1 | 2 | 3 | 4 | 5 | | 6 | Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng. | | 1 | 2 | 3 | 4 | 5 | | 7 | Trong phần kết luận, tác giả chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai. | | 1 | 2 | 3 | 4 | 5 | | **Kỹ năng viết (10)** | | | | | | | | | 8 | Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định | | 1 | 2 | 3 | 4 | 5 | | 9 | Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.) | | 1 | 2 | 3 | 4 | 5 | | **Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)** | | | | | | | | | 10a | | Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế | 5 | | | | | | 10b | | Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest. | 2 | | | | | | 10c | | Không có thành tích về nghiên cứu khoa học | 0 | | | | | | **Điểm tổng** | | | **/50** | | | | | | **Điểm tổng quy đổi về thang 10** | | |  | | | | |   ***3. Nhận xét thêm của Thầy/Cô***  ................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................................    Ngày: ... / ... / 20..  **Người nhận xét**  (Ký và ghi rõ họ tên) | |

**ĐỀ TÀI TỐT NGHIỆP**

Biểu mẫu của Đề tài/khóa luận tốt nghiệp/thực tập theo qui định của Trường, tuy nhiên cần đảm bảo giáo viên giao đề tài ký và ghi rõ họ và tên.

Trường hợp có 2 giáo viên hướng dẫn thì sẽ cùng ký tên.

Giáo viên hướng dẫn

Ký và ghi rõ họ tên

**Lời cảm ơn**

Em xin chân thành cảm ơn giảng viên hướng dẫn TS. Trần Mạnh Hoàng đã tạo điều kiện giúp đỡ em hoàn thành đồ án này. Em xin cảm ơn Trường Điện - Điện tử và đặc biệt các thành viên của Công Ty Cổ phần Công Nghệ Minastik đã tạo điều kiện giúp đỡ và hỗ trợ cho em trong suốt quá trình thực hiện đồ án tốt nghiệp. Do thời gian và kiến thức của em có hạn nên quá trình thực hiện đồ án không thể tránh khỏi những sai sót nhất định. Em rất mong muốn nhận được sự đóng góp ý kiến quý giá và có giá trị từ giáo viên và các bạn. Em xin chân thành cảm ơn.

**Tóm tắt nội dung đồ án**

Hiện nay, các mô hình ngôn ngữ lớn (LLM) như LLaMA, Mistral hay BLOOM đang đóng vai trò quan trọng trong các hệ thống chatbot, trợ lý ảo, phân tích văn bản… Tuy nhiên, để áp dụng được các mô hình này vào những bài toán thực tế tại Việt Nam, cần tiến hành tinh chỉnh (fine-tune) mô hình theo ngữ cảnh và dữ liệu tiếng Việt phù hợp. Vì vậy, đồ án này được thực hiện nhằm xây dựng một quy trình huấn luyện và triển khai chatbot dựa trên mô hình LLaMA với dữ liệu hội thoại tiếng Việt tùy chỉnh.

Đồ án xây dựng framework LlamaTuningModel, cho phép tinh chỉnh LLM bằng các phương pháp tối ưu như LoRA, QLoRA, DPO. Quá trình huấn luyện được triển khai bằng ngôn ngữ Python, sử dụng PyTorch, HuggingFace Transformers, Gradio và thực thi trên phần cứng có GPU hoặc chỉ CPU.

Đầu vào là tập dữ liệu hội thoại được định dạng JSON; đầu ra là chatbot có thể trả lời hội thoại tiếng Việt một cách tự nhiên. Kết quả cho thấy chatbot hoạt động ổn định, trả lời chính xác theo ngữ cảnh và có thể mở rộng vào các ứng dụng như tư vấn khách hàng, trợ lý học tập, v.v.

Nội dung đồ án được chia thành 6 chương:

* Chương 1: Giới thiệu tổng quan: Chương này là chương giới thiệu, đặt ra vấn đề và phát biểu ra vấn đề sẽ giải quyết trong phạm vi đồ án này
* Chương 2: Cơ sở lí thuyết và công nghệ liên quan: Chương này đưa ra bàn luận về cơ sở lý thuyết của những thuật toán hay cách làm sẽ sử dụng trong đồ án này
* Chương 3: Phân tích yêu cầu và thiết kế hệ thống: Chương này đưa ra quá trình xây dựng các mô hình, thiết kế pipeline fine‑tune và inference lại thành bài toán tinh chỉnh mô hình
* Chương 4: Cài đặt và triển khai hệ thống: Chương này trình bày quá trình cài đặt hệ thống huấn luyện và triển khai mô hình ngôn ngữ lớn (LLM) để xây dựng chatbot dựa trên nền tảng mã nguồn mở
* Chương 5: Đánh giá và kết quả thực nghiệm: Chương này đưa ra các kịch bản thực nghiệm cũng như đánh giá kết quả của bài toán
* Chương 6: Kết luận và phát triển: Chương này đưa ra kết luận của đề tài cũng như phương hướng phát triển

Sinh viên thực hiện

Ký và ghi rõ họ tên

**MỤC LỤC**

Nội dung

[CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN 14](#_Toc200485412)

[1.1 Bối cảnh và động lực 14](#_Toc200485413)

[1.2 Vấn đề cần nghiên cứu 14](#_Toc200485414)

[1.3 Mục tiêu đồ án 14](#_Toc200485415)

[1.4 Phạm vi thực hiện 14](#_Toc200485416)

[1.5 Phương pháp tiếp cận 15](#_Toc200485417)

[1.6 Cấu trúc báo cáo 15](#_Toc200485418)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ LIÊN QUAN 16](#_Toc200485419)

[2.1 Mô hình ngôn ngữ lớn (LLM) là gì? 16](#_Toc200485420)

[2.2 Giới thiệu về mô hình LLaMA 16](#_Toc200485421)

[2.3 Các kỹ thuật fine-turning LLM (LoRA, QLoRA, DPO, SPT ) 16](#_Toc200485422)

[2.4 Cấu trúc của LTM 17](#_Toc200485423)

[2.5 Kiến trúc Transformer và Attention 17](#_Toc200485424)

[2.6 Giới thiệu về Hugging Face, PEFT, TRL, v.v. 17](#_Toc200485425)

[CHƯƠNG 3. PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG 19](#_Toc200485426)

[3.1 Phân tích yêu cầu (chức năng và phi chức năng) 19](#_Toc200485427)

[3.1.1 Yêu cầu chức năng 19](#_Toc200485428)

[3.1.2 Yêu cầu phi chức năng 19](#_Toc200485429)

[3.2 Kiến trúc hệ thống tổng thể 19](#_Toc200485430)

[3.3 Mô hình luồng dữ liệu 20](#_Toc200485431)

[3.4 Thiết kế quy trình huấn luyện 20](#_Toc200485432)

[3.5 Thiết kế mô-đun phục vụ (Inference, API, UI nếu có) 21](#_Toc200485433)

[CHƯƠNG 4. CÀI ĐẶT VÀ TRIỂN KHAI HỆ THỐNG 22](#_Toc200485434)

[4.1 Môi trường thực nghiệm 22](#_Toc200485435)

[4.2 Tiền xử lý và chuẩn bị dữ liệu 22](#_Toc200485436)

[4.2.1 Định dạng dữ liệu hỗ trợ: 22](#_Toc200485437)

[4.2.2 Các bước xử lý: 22](#_Toc200485438)

[4.3 Cấu hình LTM 23](#_Toc200485439)

[4.4 Huấn luyện mô hình 23](#_Toc200485440)

[4.5 Triển khai mô hình (serving, API, web app...) 23](#_Toc200485441)

[4.5.1 Triển khai chatbot demo: 24](#_Toc200485442)

[4.5.2 Triển khai API (tùy chọn): 24](#_Toc200485443)

[4.6 Kết nối giao diện người dùng (nếu có) 24](#_Toc200485444)

[CHƯƠNG 5. ĐÁNH GIÁ VÀ KẾT QUẢ THỰC NGHIỆM 25](#_Toc200485445)

[5.1 Tiêu chí đánh giá 25](#_Toc200485446)

[5.2 Kết quả huấn luyện 25](#_Toc200485447)

[5.3 So sánh các kỹ thuật fine-tuning 26](#_Toc200485448)

[5.4 Khả năng tổng quát hóa và hiệu suất inference 26](#_Toc200485449)

[5.5 Phân tích lỗi (Error Analysis) 26](#_Toc200485450)

[CHƯƠNG 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 28](#_Toc200485451)

[6.1 Tóm tắt kết quả đạt được 28](#_Toc200485452)

[6.2 Đánh giá hạn chế 28](#_Toc200485453)

[6.3 Hướng phát triển tương lai 28](#_Toc200485454)

[TÀI LIỆU THAM KHẢO 30](#_Toc200485455)

[PHỤ LỤC 30](#_Toc200485456)

**DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT**

LlamaTuningModel LTM

**DANH MỤC HÌNH VẼ**

[Hình 2.1 Đồ thị kiểu bánh 20](#_Toc20580104)

[Hình 2.2 Đồ thị kiểu thanh ngang 20](#_Toc20580105)

[Hình 2.3 Đồ thị kiểu cột đứng 21](#_Toc20580106)

[Hình 2.4 Đồ thị kiểu đường 21](#_Toc20580107)

[Hình 2.5 Đồ thị kiểu diện tích 22](#_Toc20580108)

**DANH MỤC BẢNG BIỂU**

[Bảng 1.1 Thống kê các thiết bị và giá thành 8](#_Toc20580109)

# GIỚI THIỆU TỔNG QUAN

## Bối cảnh và động lực

Trong những năm gần đây, sự phát triển nhanh chóng của các mô hình ngôn ngữ lớn (Large Language Models – LLMs) như GPT, LLaMA, BLOOM hay Falcon đã mở ra một kỷ nguyên mới cho các ứng dụng xử lý ngôn ngữ tự nhiên (NLP). Các hệ thống chatbot hiện đại được huấn luyện từ các mô hình này có khả năng hiểu ngữ cảnh, trả lời câu hỏi phức tạp và hỗ trợ người dùng trong nhiều lĩnh vực như giáo dục, chăm sóc khách hàng, lập trình, y tế, và nhiều hơn nữa.

Tuy nhiên, việc huấn luyện và triển khai một mô hình LLM thường đòi hỏi lượng tài nguyên phần cứng rất lớn, kiến thức chuyên môn sâu về AI, và quy trình xử lý phức tạp. Điều này gây ra rào cản đối với sinh viên, nhà nghiên cứu độc lập hoặc doanh nghiệp nhỏ muốn ứng dụng công nghệ LLM vào thực tiễn.

Chính vì vậy, em quyết định xây dựng công cụ LlamaTuningModel (LTM) – giúp đơn giản hóa quá trình huấn luyện, fine-tune và triển khai các mô hình LLM mà không cần phải viết nhiều dòng mã. Đây là động lực để thực hiện đồ án này, nhằm nghiên cứu và xây dựng một hệ thống chatbot thực tiễn, từ việc huấn luyện cho đến triển khai, sử dụng các mô hình LLM trên nền tảng LTM.

## Vấn đề cần nghiên cứu

Vấn đề nghiên cứu chính của đồ án này là:

* Làm thế nào để fine-tune một mô hình LLM mã nguồn mở trên dữ liệu hội thoại tùy chọn nhằm xây dựng chatbot theo chủ đề cụ thể?
* Làm thế nào để tận dụng LTM nhằm đơn giản hóa quá trình huấn luyện, kiểm thử và triển khai chatbot?
* So sánh hiệu quả giữa các kỹ thuật fine-tuning phổ biến hiện nay (LoRA, QLoRA, SFT, DPO, v.v.)

## Mục tiêu đồ án

Mục tiêu chính của đồ án bao gồm:

* Tìm hiểu tổng quan về mô hình LLM và các kỹ thuật fine-tuning phổ biến.
* Cài đặt và cấu hình framework LTM để hỗ trợ quá trình huấn luyện.
* Chuẩn bị và xử lý dữ liệu hội thoại phù hợp với định dạng yêu cầu.
* Thực hiện fine-tuning một hoặc nhiều mô hình LLM với các kỹ thuật khác nhau.
* Đánh giá chất lượng của chatbot sau huấn luyện.
* Triển khai chatbot lên nền tảng Gradio hoặc API phục vụ người dùng.
* So sánh kết quả thu được giữa các phương pháp fine-tuning.

## Phạm vi thực hiện

* + Tập trung vào các mô hình LLM mã nguồn mở như LLaMA, BLOOM, Mistral, v.v.
  + Sử dụng dữ liệu hội thoại định dạng JSON (instruction-tuning hoặc conversation).
  + Áp dụng các kỹ thuật fine-tuning tiêu biểu như LoRA, QLoRA, SFT, DPO.
  + Triển khai thử nghiệm chatbot trên môi trường cục bộ (local) hoặc Google Colab.
  + Không tập trung xây dựng giao diện người dùng phức tạp, chỉ sử dụng Gradio hoặc API đơn giản.

## Phương pháp tiếp cận

Đồ án được triển khai theo các bước:

1. **Nghiên cứu lý thuyết**: LLM, kỹ thuật fine-tuning, kiến trúc Transformer, Attention.
2. **Tìm hiểu công cụ LTM**: Cài đặt, cấu hình, quy trình sử dụng.
3. **Tiền xử lý dữ liệu**: Chuyển đổi dữ liệu thành định dạng phù hợp để huấn luyện.
4. **Huấn luyện mô hình**: Sử dụng LTM để fine-tune LLM với cấu hình cụ thể.
5. **Đánh giá mô hình**: Dựa trên các tiêu chí định tính và định lượng (Perplexity, BLEU, v.v.)
6. **Triển khai chatbot**: Sử dụng Gradio để giao tiếp với người dùng cuối.

## Cấu trúc báo cáo

Báo cáo đồ án bao gồm 6 chương chính:

* **Chương 1: Giới thiệu tổng quan** – Trình bày tổng quan về đề tài, mục tiêu, phạm vi và phương pháp tiếp cận.
* **Chương 2: Cơ sở lý thuyết và công nghệ liên quan** – Giới thiệu về các mô hình LLM, kỹ thuật fine-tuning và các thư viện được sử dụng.
* **Chương 3: Phân tích yêu cầu và thiết kế hệ thống** – Phân tích chức năng cần có, mô hình hệ thống và luồng xử lý dữ liệu.
* **Chương 4: Cài đặt và triển khai hệ thống** – Trình bày chi tiết quá trình huấn luyện và triển khai chatbot.
* **Chương 5: Đánh giá và kết quả thực nghiệm** – Phân tích kết quả thu được, đánh giá hiệu quả mô hình.
* **Chương 6: Kết luận và hướng phát triển** – Tóm tắt kết quả và đề xuất các cải tiến trong tương lai.

# CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ LIÊN QUAN

## Mô hình ngôn ngữ lớn (LLM) là gì?

Mô hình ngôn ngữ lớn (Large Language Model – LLM) là một loại mô hình học sâu (deep learning model) được huấn luyện trên một lượng lớn văn bản nhằm hiểu và sinh ngôn ngữ tự nhiên. LLM có khả năng dự đoán từ tiếp theo trong một chuỗi, sinh câu trả lời, viết văn bản, dịch ngôn ngữ, tóm tắt, và nhiều tác vụ xử lý ngôn ngữ tự nhiên khác.

Đặc trưng của LLM:

* + Số lượng tham số lớn (từ hàng trăm triệu đến hàng trăm tỷ tham số).
  + Học biểu diễn ngữ nghĩa sâu nhờ vào các tầng attention phức tạp.
  + Có khả năng học chuyển tiếp (transfer learning) và thích nghi với tác vụ mới nhờ fine-tuning.

Một số LLM nổi bật hiện nay: GPT-3/4 (OpenAI), LLaMA (Meta), Falcon (TII), BLOOM (BigScience), Mistral…

## Giới thiệu về mô hình LLaMA

LLaMA (Large Language Model Meta AI) là dòng mô hình ngôn ngữ lớn do Meta AI phát triển và công bố năm 2023. Mô hình này được tối ưu để đạt hiệu suất cao ngay cả ở quy mô nhỏ hơn các mô hình tương đương.

Đặc điểm của LLaMA:

* Được huấn luyện trên lượng dữ liệu khổng lồ (1.4T tokens).
* Có nhiều biến thể với số lượng tham số khác nhau: 7B, 13B, 30B, 65B.
* Sử dụng kiến trúc Transformer decoder-only, tương tự GPT.
* Được thiết kế để có thể fine-tune và triển khai dễ dàng hơn so với GPT-3.

LLaMA ngày càng phổ biến trong cộng đồng mã nguồn mở nhờ khả năng mở rộng, chất lượng sinh ngôn ngữ cao và sự hỗ trợ của các thư viện như Hugging Face.

## Các kỹ thuật fine-turning LLM (LoRA, QLoRA, DPO, SPT )

Fine-tuning là quá trình điều chỉnh lại một mô hình đã được huấn luyện (pretrained) để phục vụ một tác vụ cụ thể. Các phương pháp fine-tuning phổ biến hiện nay bao gồm:

* + SFT (Supervised Fine-Tuning): Huấn luyện mô hình với tập dữ liệu gán nhãn đầu vào/đầu ra theo hướng có giám sát.
  + LoRA (Low-Rank Adaptation): Thay vì cập nhật toàn bộ trọng số mô hình, chỉ cập nhật các ma trận thấp hạng chèn vào mô hình gốc → tiết kiệm bộ nhớ và thời gian.
  + QLoRA (Quantized LoRA): Kết hợp LoRA với kỹ thuật lượng tử hóa (quantization) để giảm bộ nhớ GPU cần thiết, cho phép huấn luyện mô hình lớn trên máy tính cá nhân hoặc Google Colab.
  + DPO (Direct Preference Optimization): Tối ưu hóa mô hình dựa trên phản hồi ưu tiên từ người dùng (so sánh giữa các câu trả lời), thường dùng để tinh chỉnh độ tự nhiên của câu trả lời.

## Cấu trúc của LTM

**LTM** giúp đơn giản hóa việc fine-tuning các mô hình LLM. Thư viện này hỗ trợ nhiều kỹ thuật tinh chỉnh hiện đại như SFT, LoRA, QLoRA, DPO, PPO...

**Các thành phần chính:**

* train.py: Script chính để huấn luyện mô hình với config tùy chỉnh.
* configs/: Chứa các tệp cấu hình YAML cho từng phương pháp huấn luyện.
* data/: Hỗ trợ xử lý nhiều định dạng dữ liệu (alpaca, sharegpt, chatml…).
* src/: Chứa logic huấn luyện, định nghĩa mô hình, callback, trainer, tokenizer...
* inference/: Hỗ trợ sinh văn bản và tạo giao diện Gradio hoặc API endpoint.

LTM tương thích tốt với mô hình từ Hugging Face và tận dụng thư viện PEFT (Parameter Efficient Fine-Tuning) để hỗ trợ LoRA và QLoRA.

## Kiến trúc Transformer và Attention

Transformer là kiến trúc nền tảng cho hầu hết các LLM hiện đại. Được giới thiệu bởi Vaswani et al. (2017), Transformer sử dụng cơ chế Attention để xác định trọng số tương quan giữa các từ trong câu.

**Thành phần chính:**

* **Self-Attention:** Cho phép mô hình hiểu mối liên hệ giữa các từ trong cùng một câu.
* **Multi-head Attention:** Kết hợp nhiều self-attention song song để học được thông tin ở các không gian khác nhau.
* **Feed Forward Network:** Các lớp tuyến tính nối tiếp giúp trích xuất đặc trưng sâu hơn.
* **Positional Encoding:** Thêm thông tin vị trí vào input để mô hình hiểu thứ tự từ.

Kiến trúc này là nền tảng cho GPT, BERT, LLaMA, và hầu hết các LLM hiện nay.

## Giới thiệu về Hugging Face, PEFT, TRL, v.v.

* + Hugging Face: Nền tảng mã nguồn mở nổi tiếng hỗ trợ hàng nghìn mô hình NLP. Cung cấp kho transformers, datasets, accelerate giúp truy cập và huấn luyện LLM dễ dàng.
  + PEFT (Parameter-Efficient Fine-Tuning): Thư viện tối ưu hóa huấn luyện LLM với ít tham số hơn bằng các phương pháp như LoRA, Prefix Tuning…
  + TRL (Transformer Reinforcement Learning): Thư viện hỗ trợ huấn luyện mô hình LLM với reinforcement learning, hỗ trợ kỹ thuật như PPO, DPO dùng để fine-tune mô hình theo phản hồi người dùng.
  + Gradio: Công cụ tạo giao diện web đơn giản cho mô hình học máy, thường dùng để demo chatbot.

# PHÂN TÍCH YÊU CẦU VÀ THIẾT KẾ HỆ THỐNG

## Phân tích yêu cầu (chức năng và phi chức năng)

### Yêu cầu chức năng

Hệ thống cần đáp ứng các chức năng chính sau:

* Cho phép người dùng huấn luyện mô hình ngôn ngữ lớn (LLM) với tập dữ liệu tùy chọn.
* Hỗ trợ các kỹ thuật fine-tuning như LoRA, QLoRA, DPO.
* Cung cấp giao diện người dùng để cấu hình huấn luyện (Gradio UI).
* Tự động xử lý và kiểm tra định dạng dữ liệu đầu vào (JSON, Alpaca, ChatML...).
* Hiển thị quá trình huấn luyện và lưu mô hình sau khi hoàn tất.
* Cho phép người dùng tương tác với mô hình đã fine-tuned thông qua chatbot.
* Cung cấp endpoint hoặc giao diện demo để phục vụ mô hình.

### Yêu cầu phi chức năng

* **Hiệu năng:** Hệ thống phải tận dụng tối ưu tài nguyên phần cứng (GPU/CPU) trong quá trình huấn luyện.
* **Tính mở rộng:** Hỗ trợ dễ dàng việc thêm các mô hình hoặc dataset khác.
* **Tính khả dụng:** Cung cấp UI dễ sử dụng với cấu hình trực quan.
* **Khả năng tùy biến:** Cho phép người dùng chọn mô hình nền, số epoch, learning rate và chiến lược fine-tuning.
* **Khả năng tái sử dụng mô hình:** Mô hình sau khi fine-tuned có thể được tải lại và sử dụng cho inference.

## Kiến trúc hệ thống tổng thể

Hệ thống bao gồm các thành phần chính sau:

A screenshot of a computer program

AI-generated content may be incorrect.

## Mô hình luồng dữ liệu

1. **Người dùng nhập thông số huấn luyện:** Chọn mô hình, batch size, epoch, LoRA rank, file dữ liệu…
2. **Dữ liệu được kiểm tra định dạng:** Tự động nhận dạng và chuyển đổi về chuẩn input-output.
3. **Khởi tạo pipeline huấn luyện:** Sử dụng LTM với config tương ứng.
4. **Quá trình huấn luyện diễn ra:** Hiển thị log tiến trình trên UI.
5. **Lưu mô hình đã fine-tuned:** Có thể tải về hoặc sử dụng luôn cho inference.
6. **Tạo giao diện chat demo:** Người dùng có thể hỏi đáp với mô hình.

## Thiết kế quy trình huấn luyện

Quy trình huấn luyện được triển khai thông qua train\_web.py như sau:

1. Chọn mô hình gốc: Có thể là LLaMA, Mistral, Bloomz, v.v.

2. Lựa chọn phương pháp fine-tuning: LoRA, QLoRA, SFT, DPO…

3. Chuẩn bị dữ liệu: Hệ thống hỗ trợ tự động phân tích định dạng ChatML, Alpaca, ShareGPT, JSON chuẩn.

4. Tạo cấu hình huấn luyện: Sinh file config/train.yaml tương ứng.

5. Huấn luyện mô hình: Ghi lại log, lưu checkpoint, theo dõi loss, metric...

6. Lưu mô hình: Tự động đẩy lên Hugging Face hoặc lưu tại máy.

## Thiết kế mô-đun phục vụ (Inference, API, UI nếu có)

Sau khi huấn luyện hoàn tất, hệ thống triển khai mô hình theo các hình thức sau:

* **Gradio Chatbot (mặc định):**
  + Giao diện đơn giản cho phép người dùng nhập prompt và nhận câu trả lời.
  + Dễ triển khai và tương thích với nhiều môi trường.
* **API REST (tùy chọn):**
  + Cho phép tích hợp mô hình vào ứng dụng khác (mobile app, website…).
  + Sử dụng FastAPI hoặc Flask để tạo endpoint.
* **Tích hợp vào ứng dụng thật (tùy chọn):**
  + Hệ thống có thể được nhúng vào các trợ lý ảo, hệ thống gợi ý, hoặc chatbot trong doanh nghiệp.

# CÀI ĐẶT VÀ TRIỂN KHAI HỆ THỐNG

## Môi trường thực nghiệm

Hệ thống được triển khai trên môi trường có cấu hình như sau:

| **Thành phần** | **Thông số kỹ thuật** |
| --- | --- |
| Hệ điều hành | Ubuntu 20.04 / Google Colab / Windows WSL2 |
| GPU | NVIDIA Tesla T4 / A100 / RTX 4090 (tùy môi trường) |
| Bộ xử lý | Intel i7 / AMD Ryzen / 2 vCPU (Colab) |
| RAM | 16GB / 32GB |
| Python version | Python 3.10+ |
| CUDA version | CUDA 11.8+ |
| Thư viện chính | transformers, peft, trl, gradio, bitsandbytes, accelerate, datasets, scikit-learn, v.v. |

Lưu ý: Trong môi trường không hỗ trợ GPU, hệ thống vẫn có thể chạy huấn luyện LoRA mô hình nhỏ như TinyLLaMA, Mistral-7B (QLoRA) nhưng sẽ bị giới hạn hiệu suất.

## Tiền xử lý và chuẩn bị dữ liệu

### Định dạng dữ liệu hỗ trợ:

* Alpaca / ShareGPT / ChatML
* JSON với các key: instruction, input, output hoặc prompt, response

### Các bước xử lý:

1. Xác định định dạng đầu vào (auto detect).
2. Chuyển đổi về chuẩn của transformers.Dataset.
3. Tokenize bằng tokenizer tương ứng với mô hình (tự động load từ Hugging Face).
4. Chia nhỏ thành các đoạn phù hợp với max\_seq\_length.

Ví dụ mẫu dữ liệu:

A white background with green text

AI-generated content may be incorrect.

## Cấu hình LTM

LTM hỗ trợ tạo cấu hình trực quan trên UI hoặc bằng file YAML. Ví dụ:

A computer screen shot of a computer code

AI-generated content may be incorrect.

Người dùng có thể tạo config này bằng UI train\_web.py, không cần viết tay.

## Huấn luyện mô hình

Chạy dòng lệnh hoặc giao diện Gradio để huấn luyện:



Quy trình:

1. Lựa chọn mô hình (Bloom, LLaMA, Mistral, Falcon, etc.).
2. Cấu hình các thông số như epoch, learning rate, batch size, LoRA rank.
3. Tải dữ liệu.
4. Nhấn Start Training.
5. Theo dõi loss, accuracy, và quá trình huấn luyện trực tiếp trên UI.

Sau khi huấn luyện xong, mô hình sẽ được lưu tại thư mục ./outputs và có thể upload lên Hugging Face hub.

## Triển khai mô hình (serving, API, web app...)

### Triển khai chatbot demo:

Sau khi huấn luyện, bạn có thể sử dụng script chat\_web.py để mở chatbot Gradio:

A close-up of a text

AI-generated content may be incorrect.

Chatbot cho phép người dùng nhập câu hỏi và nhận câu trả lời từ mô hình đã fine-tuned.

### Triển khai API (tùy chọn):

Nếu không muốn dùng UI, có thể xây dựng API phục vụ bằng FastAPI:

A screenshot of a computer code

AI-generated content may be incorrect.

## Kết nối giao diện người dùng (nếu có)

Hệ thống hỗ trợ giao diện người dùng thông qua:

* **Gradio UI:** Sẵn có trong train\_web.py và chat\_web.py
* **Web frontend riêng:** Có thể kết nối API để xây dựng web chatbot thực tế.

Nếu mở rộng thành ứng dụng thật, có thể sử dụng React/Vue frontend, API backend, Redis cache và Docker cho deployment.

# ĐÁNH GIÁ VÀ KẾT QUẢ THỰC NGHIỆM

## Tiêu chí đánh giá

Để đánh giá hiệu quả của mô hình sau fine-tuning, ta sử dụng một số tiêu chí chính như sau:

| **Tiêu chí** | **Mô tả** |
| --- | --- |
| **Loss** | Hàm mất mát trong quá trình huấn luyện (CrossEntropy) |
| **Perplexity (PPL)** | Độ rối của mô hình — càng thấp càng tốt |
| **Exact Match (EM)** | Tỷ lệ câu trả lời khớp chính xác với câu chuẩn |
| **BLEU / ROUGE** | Đánh giá mức độ tương đồng giữa câu trả lời mô hình và câu chuẩn |
| **Inference latency** | Độ trễ trung bình mỗi lượt sinh văn bản |
| **Satisfaction Rate** | Tỷ lệ người dùng đánh giá phản hồi mô hình là phù hợp/trả lời đúng |

Các tiêu chí định lượng và định tính này giúp đánh giá toàn diện khả năng sinh ngôn ngữ của mô hình sau khi tinh chỉnh.

## Kết quả huấn luyện

Một số kết quả ghi nhận được trong quá trình huấn luyện mô hình với LTM như sau:

| **Mô hình** | **Dataset** | **Phương pháp fine-tune** | **Epoch** | **Train Loss** | **Eval Loss** | **Perplexity** |
| --- | --- | --- | --- | --- | --- | --- |
| LLaMA-2-7B | Alpaca-like | LoRA | 3 | 1.52 | 1.76 | 5.84 |
| Bloomz-1B1 | JSON custom | QLoRA | 5 | 1.71 | 1.88 | 6.55 |
| Mistral-7B-Instruct | ShareGPT subset | DPO + SFT | 2 | 1.48 | 1.61 | 5.00 |

**Lưu ý:** Perplexity thấp hơn cho thấy mô hình học được mối liên hệ tốt hơn giữa các token trong văn bản.

## So sánh các kỹ thuật fine-tuning

LTM hỗ trợ nhiều kỹ thuật tinh chỉnh khác nhau, dưới đây là so sánh ngắn:

| **Tiêu chí** | **LoRA** | **QLoRA** | **DPO** |
| --- | --- | --- | --- |
| GPU RAM tiêu thụ | Trung bình (~12GB) | Rất thấp (~6GB) | Cao (~24GB) |
| Khả năng tổng quát hóa | Tốt | Khá tốt | Rất tốt |
| Hỗ trợ instruction tune | Có | Có | Có (ưu việt hơn) |
| Khả năng áp dụng thực tế | Cao | Cao | Cao (phức tạp hơn) |

=> **QLoRA** phù hợp với môi trường hạn chế tài nguyên, **DPO** mạnh hơn nhưng đòi hỏi hệ thống mạnh và dữ liệu có cặp so sánh.

## Khả năng tổng quát hóa và hiệu suất inference

Sau khi huấn luyện, mô hình được kiểm thử với các câu hỏi không xuất hiện trong tập huấn luyện:

| **Câu hỏi** | **Trả lời mô hình (tóm tắt)** | **Đánh giá** |
| --- | --- | --- |
| "Tại sao bầu trời có màu xanh?" | Mô hình giải thích đúng về tán xạ Rayleigh | Tốt |
| "Việt Nam có bao nhiêu tỉnh?" | Câu trả lời đúng và cập nhật | Rất tốt |
| "Định nghĩa đạo hàm?" | Câu trả lời chính xác, có ví dụ | Rất tốt |
| "Viết chương trình in số nguyên tố?" | Sinh code Python hoàn chỉnh, có chú thích | Rất tốt |
| "Tính √2 bằng Python" | Đưa ra lời giải đúng, gọi math.sqrt(2) | Tốt |

Tốc độ phản hồi trung bình: **~1.5s/token (Gradio UI, GPU T4)**

## Phân tích lỗi (Error Analysis)

Dù kết quả tổng thể khả quan, một số lỗi vẫn xuất hiện:

* **Sinh văn bản lan man:** Với một số mô hình, mô hình có xu hướng nói "dài dòng" không cần thiết.
* **Thiếu mạch lạc trong phản hồi:** Khi gặp input lạ, mô hình đôi khi trả lời sai ngữ cảnh.
* **Quá phụ thuộc vào cấu trúc prompt:** Một số mô hình yêu cầu cấu trúc prompt rõ ràng mới hiểu đúng.

Biện pháp khắc phục:

* Sử dụng prompt template cố định (ChatTemplate, AlpacaTemplate, v.v.)
* Sử dụng dữ liệu huấn luyện đa dạng hơn, từ nhiều nguồn.
* Kết hợp thêm đánh giá từ người dùng thật.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Tóm tắt kết quả đạt được

Trong khuôn khổ đồ án, nhóm đã triển khai thành công một hệ thống chatbot sử dụng mô hình ngôn ngữ lớn (LLM) dựa trên thư viện **LTM**. Quá trình thực hiện bao gồm các bước chính:

* **Tìm hiểu và phân tích** các kiến trúc LLM, kỹ thuật fine-tuning hiện đại như **LoRA**, **QLoRA**, **DPO**,...
* **Chuẩn bị tập dữ liệu** phù hợp (các tập dạng JSON instruction tuning).
* **Cấu hình mô hình và huấn luyện** với nhiều mô hình khác nhau như LLaMA, Bloomz, Mistral,...
* **Đánh giá kết quả huấn luyện** bằng các tiêu chí như Loss, Perplexity, và phản hồi thực tế qua giao diện chatbot.
* **Triển khai thành công giao diện Gradio**, giúp người dùng tương tác trực tiếp với chatbot sau fine-tune.

Thông qua việc sử dụng các mô hình mã nguồn mở và pipeline sẵn có của LTM, đồ án cho thấy việc huấn luyện và triển khai chatbot AI không còn quá phức tạp hay yêu cầu hạ tầng quá mạnh như trước đây.

## Đánh giá hạn chế

Mặc dù hệ thống chatbot hoạt động tốt, vẫn tồn tại một số hạn chế:

* **Chất lượng phản hồi phụ thuộc lớn vào dữ liệu huấn luyện**. Nếu dữ liệu không chuẩn hoặc không đủ đa dạng, mô hình có thể học sai lệch hoặc trả lời kém chính xác.
* **Không hỗ trợ context dài (multi-turn chat) một cách hiệu quả** trong phiên bản hiện tại.
* **Hiệu suất inference còn hạn chế** khi triển khai trên GPU phổ thông như T4 hoặc CPU.
* Giao diện Gradio còn đơn giản, chưa hỗ trợ lưu lịch sử trò chuyện hoặc tùy chỉnh real-time.

## Hướng phát triển tương lai

Trong tương lai, hệ thống có thể được mở rộng theo các hướng sau:

**(1) Nâng cấp khả năng đối thoại:**

* Hỗ trợ **multi-turn conversation** (ghi nhớ các lượt chat trước).
* Áp dụng kỹ thuật **retrieval-augmented generation (RAG)** để chatbot trả lời sát dữ liệu thực tế hơn.

**(2) Tăng hiệu năng và tối ưu hóa mô hình:**

* Chuyển sang sử dụng **mô hình nhẹ hơn như Phi-2 hoặc TinyLLaMA** nếu cần tốc độ nhanh.
* Áp dụng các phương pháp **quantization** và **distillation** để rút gọn mô hình phục vụ edge device.

**(3) Phát triển ứng dụng thực tế:**

* Đóng gói chatbot thành API, triển khai trên website hoặc tích hợp với Zalo, Telegram...
* Xây dựng chatbot chuyên biệt cho các lĩnh vực cụ thể như giáo dục, chăm sóc khách hàng, tư vấn sức khỏe...

**(4) Nghiên cứu và đánh giá sâu hơn:**

* Thử nghiệm thêm nhiều kỹ thuật fine-tune hiện đại như **DPO (Direct Preference Optimization)**, **Reinforcement Learning from Human Feedback (RLHF)**.
* Thiết kế **đánh giá người dùng thực tế** để cải thiện mức độ hài lòng.

Với nền tảng đã xây dựng, đồ án không chỉ dừng lại ở một mô hình demo mà có thể mở rộng thành một hệ thống chatbot thông minh áp dụng trong nhiều lĩnh vực đời sống.

# TÀI LIỆU THAM KHẢO

 Touvron, H., Lavril, T., Izacard, G., et al. (2023). *LLaMA: Open and Efficient Foundation Language Models*. Meta AI. <https://arxiv.org/abs/2302.13971>

 Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). *QLoRA: Efficient Finetuning of Quantized LLMs*. <https://arxiv.org/abs/2305.14314>

 Hu, E. J., Shen, Y., Wallis, P., et al. (2022). *LoRA: Low-Rank Adaptation of Large Language Models*. <https://arxiv.org/abs/2106.09685>

 OpenAI (2023). *ChatGPT and GPT-4 Technical Report*. <https://openai.com/research>

 Hugging Face. (2023). *Transformers Library*. https://huggingface.co/docs/transformers

 TRL (Transformers Reinforcement Learning). <https://github.com/huggingface/trl>

 PEFT: Parameter-Efficient Fine-Tuning. <https://github.com/huggingface/peft>

 Gradio Documentation. <https://gradio.app>

 BloomZ model: https://huggingface.co/bigscience/bloomz

 Mistral-7B model: https://huggingface.co/mistralai/Mistral-7B-v0.1

 Alpaca Dataset: <https://github.com/tatsu-lab/stanford_alpaca>

# PHỤ LỤC

1. **Chi tiết số liệu thí nghiệm**

Chi tiết cấu hình môi trường thực nghiệm

* Hệ điều hành: Ubuntu 20.04 / Google Colab Pro
* Phiên bản Python: 3.10
* CUDA Toolkit: 11.8
* GPU sử dụng:
  + Google Colab: NVIDIA Tesla T4 (16GB VRAM)
  + Local (nếu có): NVIDIA RTX 3090 hoặc A100
* Thư viện Python chính:
  + transformers, peft, datasets, accelerate, gradio, torch, scipy, scikit-learn, numpy, pandas
* RAM tối thiểu khuyến nghị: 16GB

1. **Chi tiết các bước tính toán**

Danh sách các tệp mã nguồn chính

1. train\_web.py: Tạo giao diện Gradio để người dùng cấu hình thông số huấn luyện mô hình.
2. src/train/train.py: Pipeline huấn luyện chính, xử lý dữ liệu, forward/backward pass, lưu checkpoint.
3. src/serve/gradio\_chat.py: Giao diện trò chuyện sau khi huấn luyện, triển khai mô hình inference.
4. src/utils/config.py: Tải và quản lý cấu hình huấn luyện như model\_name, dataset\_path, hyperparameters,...
5. dataset/convert\_dataset.py: Script chuyển đổi dữ liệu về định dạng JSON Instruction (Input, Output).
6. **Chi tiết sơ đồ mô phỏng**

Dữ liệu huấn luyện và kết quả thử nghiệm

Trình phụ lục tại đây:

1. Dataset sử dụng

* Dữ liệu chính:
  + Định dạng Alpaca-style (Instruction, Input, Output)
* Các bộ dữ liệu huấn luyện thử:
  + alpaca\_gpt4\_vi.json
  + finance\_qa.json *(tự tạo theo chủ đề tài chính)*
  + med\_qa.json *(tự tạo theo chủ đề y tế)*
  + Bộ dữ liệu tùy chỉnh do nhóm tự tổng hợp

2. Kết quả thử nghiệm

* Loss Curve:
  + Loss giảm đều qua từng epoch
  + Mô hình hội tụ nhanh ở epoch thứ 3 khi sử dụng LoRA
  + Log huấn luyện được lưu tại: output\_dir/training\_logs.json
* Bảng kiểm thử mô hình

| Câu hỏi | Mô hình trả lời | Ghi chú |
| --- | --- | --- |
| Tóm tắt đoạn văn sau… | ✅ Trả lời đúng cú pháp | alpaca\_gpt4\_vi |
| Khi nào nên đầu tư chứng khoán? | ✅ Trả lời phù hợp | finance\_qa.json |
| Bệnh sốt xuất huyết lây qua đâu? | ✅ Trả lời chính xác | med\_qa.json |