ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

---□&□---

BÁO CÁO ĐỒ ÁN

Project 3

Đề tài: Xây dựng mô hình phân loại tài liệu đa ngôn ngữ

GVHD: Tạ Duy Hoàng

Sinh viên thực hiện:
Nguyễn Việt Anh 20215307

Hà Nội, tháng 1 năm 2026

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành nhất tới thầy **Tạ Duy Hoàng**, người đã trực tiếp hướng dẫn, tận tình chỉ bảo và đưa ra những định hướng quý báu trong suốt quá trình em thực hiện đồ án Project 3 này. Những kiến thức và kinh nghiệm thực tế thầy chia sẻ đã giúp em vượt qua nhiều thách thức kỹ thuật, đặc biệt là trong việc tối ưu hóa các mô hình học máy hiện đại.

Em cũng xin gửi lời tri ân tới các thầy cô giáo trường Công nghệ Thông tin và Truyền thông – Đại học Bách khoa Hà Nội, những người đã truyền dạy cho em nền tảng tri thức vững chắc trong những năm học qua.

Cuối cùng, em xin cảm ơn các thầy cô và bạn bè đã luôn ủng hộ, động viên em hoàn thành tốt đồ án này. Dù đã dành nhiều tâm huyết, song báo cáo chắc chắn không tránh khỏi những thiếu sót, em rất mong nhận được những ý kiến đóng góp của các thầy cô để sản phẩm ngày càng hoàn thiện hơn.

Em xin chân thành cảm ơn!

PHẦN 1: GIỚI THIỆU TỔNG QUAN

1.1. Bối cảnh bài toán

Trong thời đại chuyển đổi số, dữ liệu dưới dạng văn bản số hóa đang tăng trưởng với tốc độ chóng mặt. Trong số đó, định dạng PDF (Portable Document Format) đã trở thành tiêu chuẩn vàng cho việc lưu trữ và trao đổi tài liệu nhờ khả năng giữ nguyên định dạng trên mọi nền tảng. Tuy nhiên, sự tiện lợi này cũng đi kèm với một thách thức lớn trong quản lý dữ liệu: **Sự đa dạng về ngôn ngữ**.

Các doanh nghiệp và tổ chức giáo dục hiện nay thường xuyên phải đối mặt với các kho lưu trữ khổng lồ chứa hàng chục nghìn tài liệu từ nhiều quốc gia khác nhau. Việc phân loại và sắp xếp các tài liệu này một cách thủ công không chỉ gây lãng phí nguồn lực nhân sự mà còn tiềm ẩn rủi ro sai sót rất cao, dẫn đến khó khăn trong việc tra cứu và khai thác thông tin sau này.

1.2. Tầm quan trọng của việc nhận diện ngôn ngữ tự động

Nhận diện ngôn ngữ (Language Identification) là bước tiền đề và cốt yếu trong mọi quy trình xử lý ngôn ngữ tự động (NLP). Nếu bước đầu tiên không xác định chính xác ngôn ngữ, các hệ thống tích hợp gồm các tiện ích và công cụ liên quan phía sau của các hệ thống lớn như:

- **Trích xuất thông tin (Information Extraction):** Sẽ sử dụng sai bộ quy tắc ngữ pháp.
- **Dịch máy (Machine Translation):** Sẽ không thể xác định ngôn ngữ nguồn và thực hiện dịch thuật ngôn ngữ.
- **Tìm kiếm thông minh (Intelligent Search):** Sẽ trả về các kết quả không liên quan do sai lệch về từ điển.

Đặc biệt, đối với các quốc gia thuộc khu vực Đông Á như Nhật Bản và Hàn Quốc, tài liệu thường chứa các ký tự tượng hình và biểu âm phức tạp (Kanji, Kana, Hangul), đòi hỏi các bộ mã hóa (encoding) chuyên biệt. Việc xây dựng một công cụ có thể tự động "đọc" và "hiểu" ngôn ngữ của file PDF ngay từ bước đầu vào là nhu cầu cấp thiết để xây dựng các hệ thống quản trị nội dung thông minh.

1.3. Mục tiêu của dự án "Hệ thống phân loại ngôn ngữ tự động”

Dự án được thực hiện với mục tiêu xây dựng một sản phẩm hoàn thiện, có khả năng tự động hóa quy trình phân loại ngôn ngữ cho các tệp PDF đa quốc gia. Sản phẩm tập trung giải quyết các bài toán cụ thể sau:

- **Hỗ trợ đa ngôn ngữ đặc thù:** Tập trung vào 4 ngôn ngữ chính có nhu cầu cao ở trong nước nói riêng và các khu vực châu Á, quốc tế nói chung: Tiếng Việt (VN), Tiếng Nhật (JP), Tiếng Hàn (KR) và Tiếng Anh (US).
- **Xử lý quy mô dữ liệu lớn:** Hệ thống được huấn luyện và kiểm thử trên tập dữ liệu gồm gần 10,000 tập tin PDF với chủ đề đa dạng thực tế.
- **Vượt qua giới hạn độ dài:** Khác với các mô hình nhận diện văn bản ngắn, sản phẩm này hướng tới xử lý các tài liệu dài và phức tạp thông qua chiến lược phân mảnh (Chunking Strategy).
- **Tối ưu hóa tài nguyên:** Thiết kế để có thể vận hành ổn định trên các cấu hình phần cứng phổ thông (như GPU NVIDIA RTX 3050 với 4GB VRAM) mà vẫn đảm bảo độ chính xác vượt trội.

1.4. Phạm vi và Đối tượng sử dụng

- **Phạm vi kỹ thuật:** Tập trung vào các file PDF dạng văn bản (text-based). Hệ thống thực hiện trích xuất nội dung văn bản, phân tích đặc trưng ngôn ngữ bằng Deep Learning và đưa ra kết quả phân loại cùng độ tin cậy tương ứng.
- **Đối tượng có thể sử dụng hệ thống:** Các đơn vị lưu trữ tài liệu, các công ty dịch thuật, bộ phận văn phòng của các tập đoàn đa quốc gia và các nhà phát triển muốn tích hợp module nhận diện ngôn ngữ vào hệ thống quản lý tài liệu (DMS) hiện có cũng như làm bước đầu chính xác cho các công cụ và tiện ích phía sau của một hệ thống lớn.

PHẦN 2: CƠ SỞ LÝ THUYẾT VÀ LỰA CHỌN MÔ HÌNH

2.1. Bài toán Nhận diện Ngôn ngữ (Language Identification –LID)

Nhận diện ngôn ngữ là bài toán phân loại văn bản (Text Classification) ở mức độ cơ bản nhưng lại đóng vai trò quyết định trong các hệ thống xử lý ngôn ngữ tự động. Mục tiêu là gán một nhãn ngôn ngữ cho một đoạn văn bản cho trước.

Trong dự án này, bài toán được định nghĩa là phân loại đa lớp (Multi-class Classification) với 4 nhãn mục tiêu:

- **vi**: Tiếng Việt (Vietnamese)
- **en**: Tiếng Anh (English)
- **ja**: Tiếng Nhật (Japanese)
- **kr**: Tiếng Hàn (Korean)

Thách thức lớn nhất nằm ở việc xử lý đồng thời hai hệ thống chữ viết: chữ Latinh (Anh, Việt) và chữ tượng hình/biểu âm (Nhật, Hàn).

2.2. Sự tiến hóa của các phương pháp LID

Để hiểu tại sao hệ thống sử dụng Deep Learning, chúng ta cần nhìn lại quá trình phát triển của các phương pháp nhận diện ngôn ngữ:

1. **Phương pháp dựa trên quy tắc (Rule-based):** Sử dụng các bộ từ điển hoặc danh sách ký tự đặc trưng. Tuy nhiên, phương pháp này thất bại khi gặp các từ vay mượn hoặc văn bản chuyên ngành.
2. **Phương pháp thống kê (n-gram):** Sử dụng tần suất xuất hiện của các cụm ký tự liên tiếp. Đây là phương pháp cốt lõi của thư viện FastText. Dù nhanh, nhưng n-gram thiếu khả năng hiểu ngữ cảnh sâu sắc.
3. **Phương pháp dựa trên Deep Learning (Transformers):** Sử dụng cơ chế chú ý (Attention) để học các đặc trưng ngôn ngữ ở mức trừu tượng cao, cho phép nhận diện chính xác ngay cả với các đoạn văn bản ngắn hoặc chứa nhiều nhiễu.

2.3. Kiến trúc Transformer và cơ chế Self-Attention

Mô hình được lựa chọn cho dự án dựa trên kiến trúc Transformer – một bước đột phá trong NLP giúp loại bỏ sự phụ thuộc vào các mạng đệ quy (RNN/LSTM).

- **Cơ chế Self-Attention:** Cho phép mô hình tập trung vào các phần khác nhau của câu để hiểu mối quan hệ giữa các từ, không phụ thuộc vào khoảng cách giữa chúng. Điều này đặc biệt quan trọng với tiếng Nhật và Hàn, nơi cấu trúc câu thường khác xa so với tiếng Anh.
- **Kiến trúc Encoder-only:** Khác với các mô hình dịch máy (Encoder-Decoder), các mô hình phân loại như XLM-RoBERTa chỉ sử dụng phần Encoder để trích xuất các vector biểu diễn đặc trưng (embeddings) tốt nhất từ văn bản đầu vào.

2.4. Mô hình XLM-RoBERTa

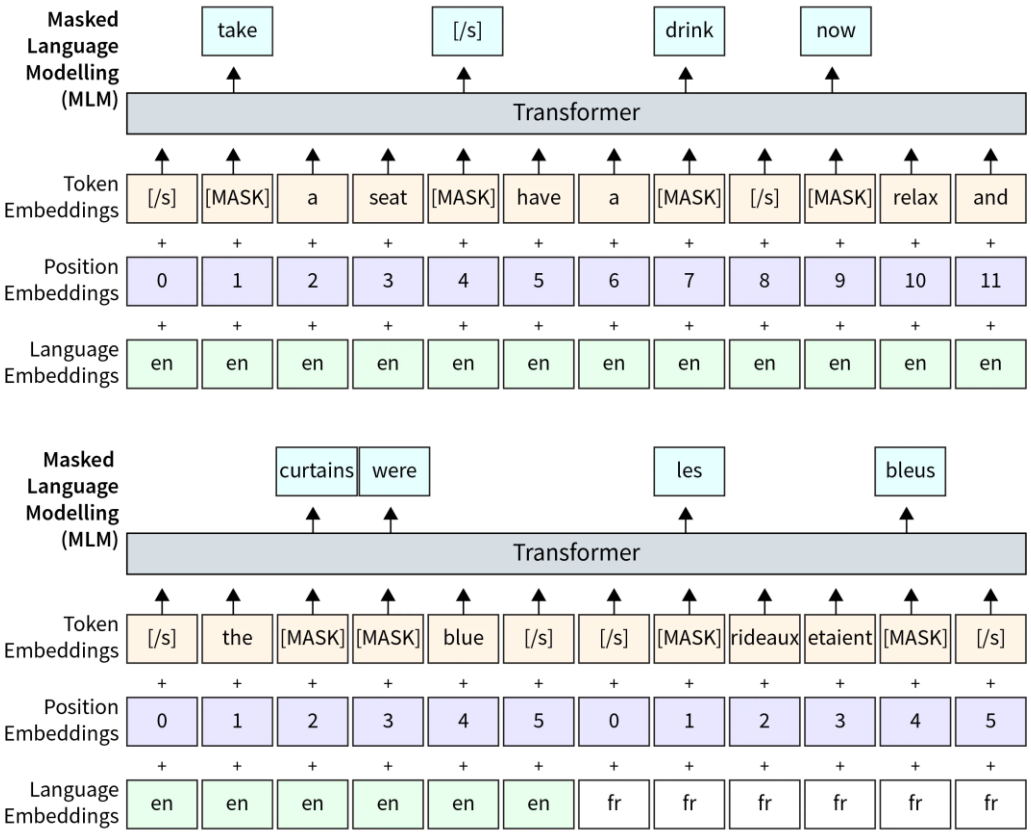
Dự án sử dụng phiên bản **XLM-RoBERTa Base** làm mô hình cốt lõi. Đây là sự kết hợp giữa kiến trúc RoBERTa mạnh mẽ và phương pháp huấn luyện đa ngôn ngữ XLM.

2.4.1. Tổng quan kiến trúc XLM

Các thành phần chính của kiến trúc XLM bao gồm:

- **Bộ mã hóa Transformer:**
  XLM sử dụng bộ mã hóa Transformer đa lớp, tương tự như BERT. Kiến trúc Transformer cho phép xử lý song song hiệu quả dữ liệu tuần tự và nắm bắt các mối quan hệ phụ thuộc tầm xa trong văn bản.
- **Mô hình ngôn ngữ che khuất đa ngôn ngữ (XLM-MLM):**
  Trong giai đoạn huấn luyện trước, XLM sử dụng một biến thể của mục tiêu mô hình ngôn ngữ che khuất (MLM) được gọi là MLM đa ngôn ngữ . Trong quá trình này, các từ từ các ngôn ngữ khác nhau được che khuất ngẫu nhiên trong một câu, và mô hình được huấn luyện để dự đoán các từ bị che khuất đó. Điều này khuyến khích mô hình học các biểu diễn ngữ cảnh không phụ thuộc vào ngôn ngữ, giúp nó hiểu nhiều ngôn ngữ một cách hiệu quả.
- **Mục tiêu song ngữ:**

XLM giới thiệu mục tiêu song ngữ, trong đó dữ liệu song song (các câu trong hai ngôn ngữ có cùng nghĩa) được tận dụng trong quá trình huấn luyện. Điều này cho phép mô hình học được sự tương ứng giữa các ngôn ngữ và chuyển giao kiến thức từ ngôn ngữ này sang ngôn ngữ khác.



Hình 1: Kiến trúc mô hình xlm.

XLM cung cấp một số tính năng và ưu điểm chính:

- **Khả năng đa ngôn ngữ:**
  XLM có thể hiểu và tạo văn bản bằng nhiều ngôn ngữ. Điều này vượt qua những hạn chế của các mô hình ngôn ngữ truyền thống, vốn được thiết kế cho các ngôn ngữ cụ thể và thiếu khả năng chuyển đổi giữa các ngôn ngữ.
- **Học chuyển giao đa ngôn ngữ (Cross-Lingual Transfer Learning - XLM):**
  Một trong những ưu điểm đáng kể của XLM là khả năng chuyển giao kiến thức đã học từ ngôn ngữ này sang ngôn ngữ khác. Điều này đặc biệt

có giá trị đối với các ngôn ngữ có dữ liệu huấn luyện hạn chế, vì mô hình có thể tận dụng thông tin từ các ngôn ngữ có liên quan.

- **Cải thiện khả năng biểu diễn đa ngôn ngữ:**
  Kiến trúc XLM, với mục tiêu song ngữ và mô hình đa ngôn ngữ (MLM), khuyến khích mô hình học các biểu diễn nắm bắt các đặc điểm không phụ thuộc vào ngôn ngữ. Điều này giúp cải thiện khả năng hiểu đa ngôn ngữ và hiệu suất trong nhiều tác vụ xử lý ngôn ngữ tự nhiên (NLP) tiếp theo.

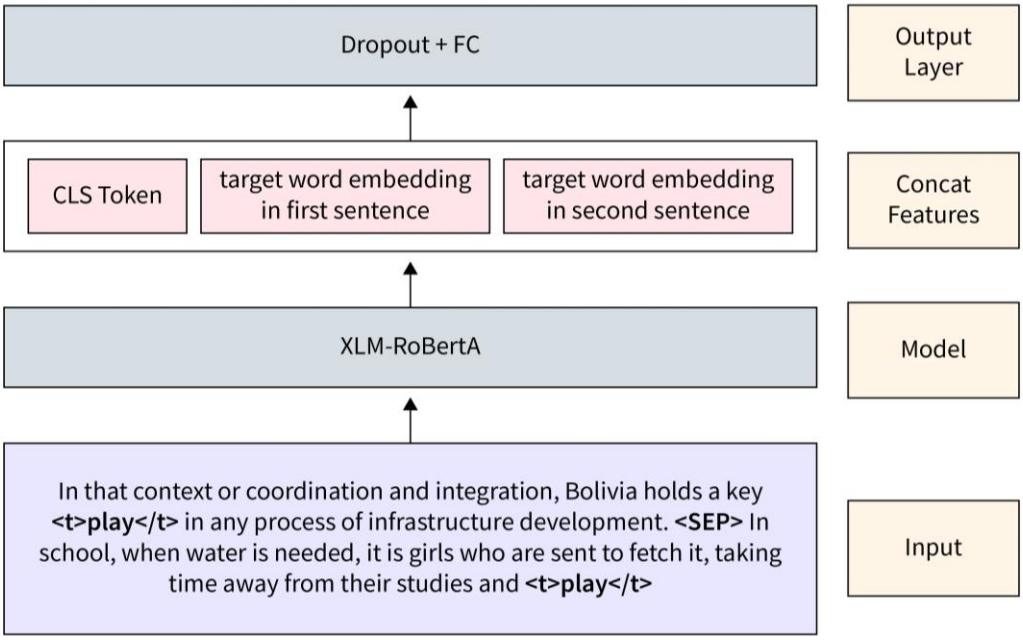
Nhìn chung, kiến trúc và mục tiêu huấn luyện của XLM biến nó thành một bước tiến quan trọng trong việc phát triển các mô hình ngôn ngữ đa ngôn ngữ, thúc đẩy sự hiểu biết xuyên ngôn ngữ và chuyển giao kiến thức trong xử lý ngôn ngữ tự nhiên.

2.4.2. Tại sao chọn XLM-RoBERTa?

XLM-RoBERTa là một phần mở rộng của XLM sử dụng kiến trúc RoBERTa , một biến thể của mô hình Transformer, để huấn luyện trước.

Nó dựa trên kiến trúc RoBERTa , một biến thể được tối ưu hóa của BERT (Bidirectional Encoder Representations from Transformers). XLM-RoBERTa xây dựng dựa trên những điểm mạnh của XLM đồng thời kết hợp những cải tiến từ RoBERTa để đạt được hiệu suất tốt hơn nữa trong hiểu ngôn ngữ đa ngôn ngữ và các tác vụ xử lý ngôn ngữ tự nhiên (NLP) tiếp theo.

Ý tưởng cốt lõi đằng sau XLM-RoBERTa là tận dụng phương pháp huấn luyện trước của RoBERTa, bao gồm huấn luyện trước quy mô lớn và tinh chỉnh siêu tham số mở rộng, để tạo ra một mô hình ngôn ngữ đa ngôn ngữ mạnh mẽ và hiệu quả hơn.



Hình 2: Mô hình XLM-RoBERTa

Kiến trúc XLM-RoBERTa bao gồm:

- **Lớp nhúng:**
  Tương tự như các mô hình transformer khác, XLM-RoBERTa bắt đầu với các lớp nhúng. Các lớp này ánh xạ các token đầu vào thành các biểu diễn vector liên tục, được gọi là nhúng từ. XLM-RoBERTa sử dụng mã hóa cặp byte (BPE) để xử lý các đơn vị dưới dạng từ, cho phép nó xử lý các từ nằm ngoài từ vựng và nắm bắt thông tin hình thái học.
- **Bộ mã hóa Transformer:**
  Cốt lõi của kiến trúc XLM-RoBERTa là bộ mã hóa Transformer. Nó bao gồm nhiều lớp cơ chế tự chú ý và mạng nơ-ron truyền thẳng. Mỗi lớp trong bộ mã hóa xử lý chuỗi đầu vào song song, cho phép mô hình nắm bắt cả các phụ thuộc cục bộ và toàn cục.
  Cơ chế tự chú ý cho phép mô hình tập trung vào các phần khác nhau của chuỗi đầu vào trong khi mã hóa thông tin ngữ cảnh. Nó tính toán trọng số chú ý cho mỗi token đầu vào, cho phép mô hình tập trung vào thông tin liên quan trong quá trình mã hóa.
  Các mạng nơ-ron truyền thẳng bên trong mỗi lớp biến đổi giúp nắm bắt các mối quan hệ phức tạp và tính phi tuyến tính trong chuỗi đầu vào.
- **Cấu trúc xử lý tiếp theo:**

Đầu ra của bộ mã hóa Transformer được truyền qua một cấu trúc xử lý tiếp theo, cấu trúc này có thể thay đổi tùy thuộc vào nhiệm vụ cụ thể mà mô hình được huấn luyện. Cấu trúc này thường bao gồm các lớp bổ sung (ví dụ: các lớp kết nối đầy đủ) để chuyển đổi các biểu diễn được mã hóa thành đầu ra dành riêng cho nhiệm vụ.

Nhìn chung, kiến trúc XLM-RoBERTa được thiết kế để học các biểu diễn câu mạnh mẽ có khả năng nắm bắt thông tin ngữ nghĩa và cú pháp trên nhiều ngôn ngữ khác nhau. Bằng cách huấn luyện trên một lượng lớn dữ liệu đa ngôn ngữ, XLM-RoBERTa có thể tận dụng thông tin chung giữa các ngôn ngữ để cải thiện hiệu suất trên nhiều nhiệm vụ đa ngôn ngữ khác nhau.

Điểm mạnh của XLM-RoBERTa:

1. **Quy mô huấn luyện khổng lồ:** Mô hình được huấn luyện trên tập dữ liệu CommonCrawl với hơn 2.5TB văn bản sạch, bao gồm hơn 100 ngôn ngữ khác nhau. Cả 4 ngôn ngữ mục tiêu (VN, JP, KR, US) đều nằm trong nhóm ngôn ngữ được hỗ trợ tốt nhất của mô hình này.
2. **Hỗ trợ đa mã hóa (SentencePiece Tokenizer):** XLM-RoBERTa sử dụng kỹ thuật Byte-Pair Encoding (BPE) trực tiếp trên dòng dữ liệu thô. Điều này cho phép mô hình xử lý bất kỳ ngôn ngữ nào mà không cần các bộ tách từ (segmentation) riêng biệt cho tiếng Nhật hay Hàn.
3. **Mục tiêu huấn luyện:** Chủ yếu sử dụng **Masked Language Modeling (MLM)** trên dữ liệu đa ngữ mà không cần dữ liệu song ngữ (unsupervised).
4. **Hiệu quả với tiếng Việt:** XLM-R cho thấy kết quả rất ấn tượng trên các bộ dữ liệu tiếng Việt, đôi khi vượt qua cả các mô hình chuyên biệt cho tiếng Việt như PhoBERT trong một số tác vụ nhất định như trả lời câu hỏi (QA)...
5. **Vượt qua "Lời nguyền đa ngữ":** Mô hình giải quyết bài toán đánh đổi giữa số lượng ngôn ngữ và độ chính xác bằng cách tăng quy mô tham số ô hình (capacity), giúp duy trì hiệu suất cao ngay cả khi số lượng ngôn ngữ tăng lên.

2.4.3. Thông số kỹ thuật của mô hình

- **Số lượng tham số:** Khoảng 270 triệu tham số (270M parameters).

- **Kích thước bộ nhớ:** ~560MB.
- **Tokenizer:** SentencePiece (hỗ trợ sub-word, giúp giảm thiểu lỗi từ lạ - Out-of-Vocabulary).
- **Độ dài chuỗi tối đa:** 512 tokens.

2.5. Lý do lựa chọn tham số và cấu hình Fine-tuning

Trong dự án này, chúng ta không huấn luyện mô hình từ đầu mà thực hiện **Fine-tuning** (tinh chỉnh).

- **Lựa chọn Fine-tuning:** Việc huấn luyện lại hoàn toàn một mô hình như XLM-RoBERTa đòi hỏi hàng nghìn GPU giờ. Bằng cách Fine-tuning, chúng ta kế thừa được tri thức ngôn ngữ từ hàng Terabyte dữ liệu và chỉ cần tinh chỉnh lớp phân loại cuối cùng (Classification Head) để phù hợp với 4 nhãn mục tiêu.
- **Lý do chọn cấu hình Base thay vì Large:** Do giới hạn phần cứng là GPU RTX 3050 (4GB VRAM), phiên bản "Base" là sự cân bằng hoàn hảo giữa hiệu năng và tài nguyên. Phiên bản "Large" (550M+ tham số) sẽ gây lỗi tràn bộ nhớ (OOM) trong quá trình huấn luyện.

PHẦN 3: QUY TRÌNH TIỀN XỬ LÝ DỮ LIỆU VÀ KỸ THUẬT TRÍCH XUẤT VĂN BẢN

Quy trình xử lý dữ liệu đóng vai trò quyết định đến chất lượng của mô hình học máy. Đối với văn bản trích xuất từ PDF, thách thức không chỉ nằm ở việc lấy được chữ mà còn là việc bảo toàn ngữ nghĩa và các đặc trưng ngôn ngữ đặc thù của 4 quốc gia: Việt Nam, Nhật Bản, Hàn Quốc và Hoa Kỳ.

3.1. Phân tích và lựa chọn công cụ trích xuất PDF

Trong quá trình phát triển, dự án đã tiến hành thử nghiệm song song hai thư viện phổ biến nhất hiện nay để đưa ra lựa chọn tối ưu cho sản phẩm cuối cùng.

3.1.1. Thử nghiệm với pdfminer.six

Ban đầu, hệ thống sử dụng **pdfminer.six** để trích xuất văn bản. Ưu điểm của thư viện này là khả năng phân tích cấu trúc layout khá chi tiết. Tuy nhiên, qua thực nghiệm trên tập dữ liệu đa ngôn ngữ, **pdfminer** bộc lộ một số hạn chế:

- **Tốc độ:** Xử lý các tệp PDF lớn chậm, gây tắc nghẽn khi xử lý batch hàng nghìn tệp.
- **Lỗi tiếng Việt:** Thường gặp vấn đề với các font chữ VNI hoặc lỗi tách dấu (ví dụ: "ên" bị tách thành "e" và "^n").

3.1.2. Chuyển đổi sang PyMuPDF (Fitz) - Lựa chọn tối ưu

Sau giai đoạn thử nghiệm, dự án đã chuyển đổi hoàn toàn sang sử dụng **PyMuPDF** (thư viện fitz). Đây là quyết định mang tính chiến lược cho sản phẩm nhờ các đặc điểm:

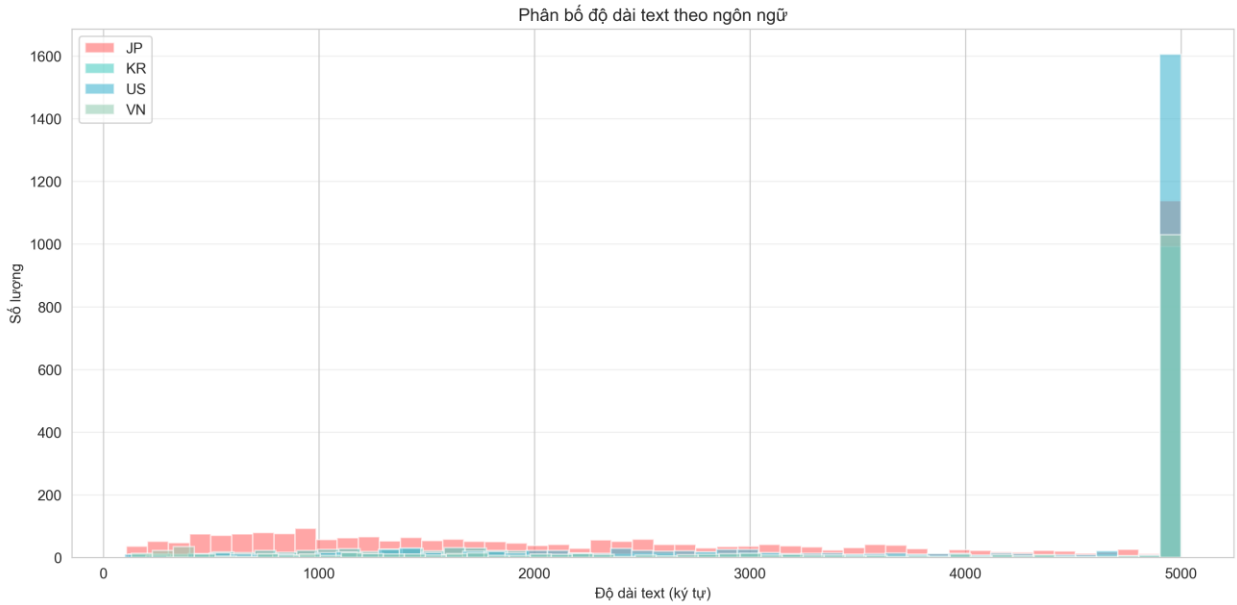
- **Hiệu năng vượt trội:** Tốc độ trích xuất nhanh hơn gấp 5-10 lần so với các thư viện dựa trên Python thuần túy.
- **Độ chính xác cao:** Xử lý cực tốt các bảng mã phức tạp của tiếng Nhật (Shift-JIS, EUC-JP) và tiếng Hàn (EUC-KR).
- **Bảo toàn ngữ nghĩa:** Hỗ trợ cơ chế TEXT_PRESERVE_LIGATURES và TEXT_PRESERVE_WHITESPACE, giúp giữ đúng khoảng cách giữa các từ, điều cực kỳ quan trọng cho việc Tokenize sau này.

3.2. Quy trình làm sạch và chuẩn hóa văn bản (Text Cleaning & Normalization)

Văn bản sau khi trích xuất từ PDF thường chứa nhiều "rác" hệ thống. Hệ thống đã triển khai bộ lọc 4 lớp:

1. **Chuẩn hóa Unicode NFKC:** Sử dụng unicodedata.normalize ("NFKC", text) để hợp nhất các tổ hợp phím và ký tự đặc biệt. Bước này giúp các ký tự tiếng Việt có dấu được đưa về dạng thống nhất, tránh việc mô hình hiểu sai cùng một từ nhưng khác bộ mã.
2. **Chuẩn hóa khoảng trắng:** Loại bỏ các ký tự xuống dòng (\n), tab (\t) và các khoảng trắng dư thừa, đưa văn bản về dạng chuỗi liên tục để mô hình Transformer dễ dàng nắm bắt ngữ cảnh.

3. **Lọc dữ liệu rác:** Tự động loại bỏ các tệp tin có dung lượng text quá ngắn (dưới 100 ký tự). Những tệp này thường chỉ chứa số trang, tiêu đề hoặc là các tệp scan không chứa lớp text ẩn (OCR-needed), nếu giữ lại sẽ làm nhiễu quá trình huấn luyện.
4. **Giới hạn ký tự:** Để đảm bảo hiệu năng huấn luyện, hệ thống giới hạn 5,000 ký tự đầu tiên cho mỗi tệp trong giai đoạn tạo Dataset.



Hình 3: Biểu đồ Phân bố Độ dài Văn bản (Text Length Distribution)

Biểu đồ Histogram thể hiện mật độ tập trung của độ dài văn bản (tính theo ký tự). Có thể thấy phần lớn tài liệu tập trung ở ngưỡng từ 3,000 đến 5,000 ký tự. Việc tập trung dữ liệu ở ngưỡng dài giúp mô hình XLM-RoBERTa tiếp cận được nhiều ngữ cảnh hơn, hỗ trợ tốt cho chiến lược Chunking ở các bước sau.

3.3. Chiến lược phân đoạn văn bản nâng cao (Enhanced Chunking Strategy)

Đây là tính năng cốt lõi giúp hệ thống xử lý được các tài liệu PDF dài lên tới 50,000 ký tự (vượt xa giới hạn 512 tokens của mô hình BERT/RoBERTa thông thường).

Chiến lược được triển khai như sau:

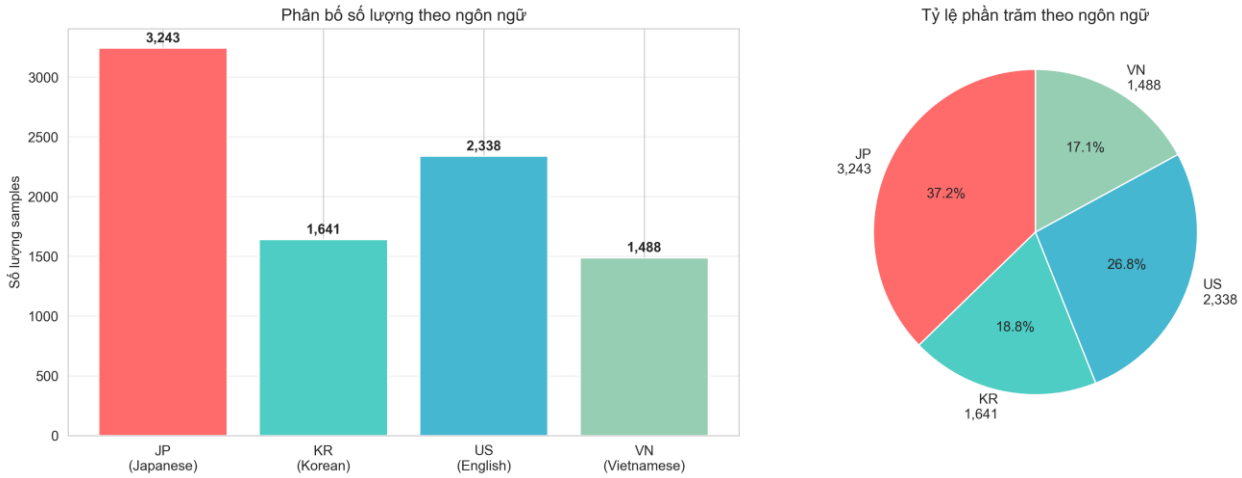
- **Chia nhỏ (Sliding Window):** Văn bản dài được chia thành các "chunk" nhỏ, mỗi chunk khoảng 2,000-2,500 ký tự.

- **Cơ chế Overlap:** Giữa các chunk có một khoảng chồng lấp 200 ký tự. Điều này đảm bảo rằng các câu văn nằm ở ranh giới giữa hai chunk không bị cắt đôi một cách đột ngột, giúp mô hình giữ được ngữ cảnh toàn vẹn.
- **Cắt theo khoảng trắng:** Hệ thống không cắt máy móc theo số ký tự mà chủ động tìm khoảng trắng gần nhất để cắt, đảm bảo tính nguyên vẹn của từ ngữ.

3.4. Thống kê bộ dữ liệu và Chia tách (Dataset Splitting)

Kết quả sau quy trình xử lý dữ liệu đã tạo ra một bộ Dataset sạch và cân bằng:

- **Tổng số mẫu thành công:** 8,721 mẫu (từ 9,686 file thô ban đầu).
- **Phân bổ:**
    Tiếng Nhật (JP): 3,247 mẫu (37.2%).
    Tiếng Anh (US): 2,338 mẫu (26.8%).
    Tiếng Hàn (KR): 1,648 mẫu (18.9%).
    Tiếng Việt (VN): 1,488 mẫu (17.1%).



Hình 4: Biểu đồ phân bố số lượng ngôn ngữ

Biểu đồ hình cột và hình tròn cho thấy sự phân bổ dữ liệu giữa 4 ngôn ngữ. Mặc dù có sự chênh lệch (tiếng Nhật chiếm tỷ trọng cao nhất với 37.2%), nhưng số lượng mẫu của mỗi lớp đều đạt ngưỡng trên 1,400 mẫu, đủ để mô hình học được các đặc trưng riêng biệt mà không bị mất cân bằng quá mức.

Hệ thống thực hiện chia dữ liệu theo phương pháp **Stratified Split** (Chia tách phân tầng) với tỷ lệ **70% Train (6,104 mẫu) - 15% Validation (1,308 mẫu) - 15% Test (1,309 mẫu)**. Việc sử dụng Stratified đảm bảo tỷ lệ các ngôn ngữ

trong tập Test giống hệt tập Train, giúp kết quả đánh giá mô hình khách quan và chính xác nhất.

PHẦN 4: THIẾT LẬP HUẤN LUYỆN VÀ TỐI ƯU HÓA MÔ HÌNH

Quá trình huấn luyện mô hình được thực hiện với chiến lược tối ưu hóa tài nguyên phần cứng, đảm bảo mô hình XLM-RoBERTa-Base (với hơn 270 triệu tham số) có thể hoạt động ổn định trên GPU có dung lượng VRAM hạn chế (4GB).

4.1. Môi trường và Cấu hình phần cứng

Dự án được triển khai trên hệ thống máy tính cục bộ với cấu hình:

- **Hệ điều hành:** Windows 11.
- **CPU:** Intel Core i7.
- **GPU:** NVIDIA RTX 3050Ti (4GB VRAM) tích hợp CUDA 12.5.
- **Thư viện lõi:** PyTorch 2.1.0 và Hugging Face Transformers.

4.2. Thiết lập tham số huấn luyện (Hyperparameters)

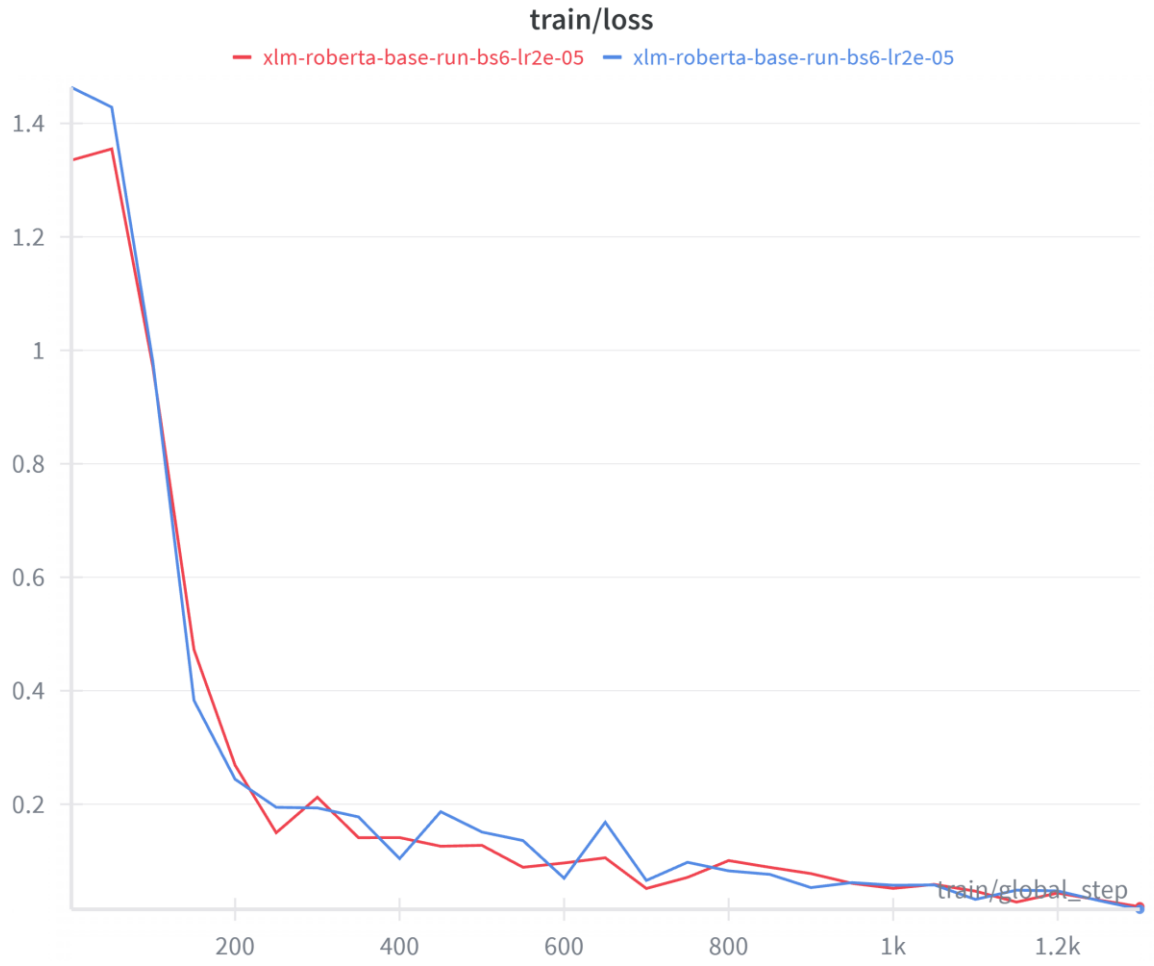
Dựa trên thực nghiệm, bộ tham số sau đã được lựa chọn để cân bằng giữa tốc độ hội tụ và độ ổn định của Gradient:

- **Số Epochs:** 3 (Đảm bảo mô hình học đủ sâu mà không gây Overfitting).
- **Batch Size per Device:** 6 (Tối ưu cho VRAM 4GB).
- **Gradient Accumulation Steps:** 2 (Tạo ra Effective Batch Size là 12, giúp ổn định quá trình cập nhật trọng số).
- **Learning Rate:** 2e-5 với cơ chế Linear Warmup (giúp mô hình không bị "gradient shock" ở những bước đầu).
- **Max Sequence Length:** 512 tokens.

4.3. Giám sát quá trình huấn luyện qua WandB

Toàn bộ quá trình huấn luyện được giám sát theo thời gian thực thông qua Dashboard của **WandB (Weights & Biases)**.

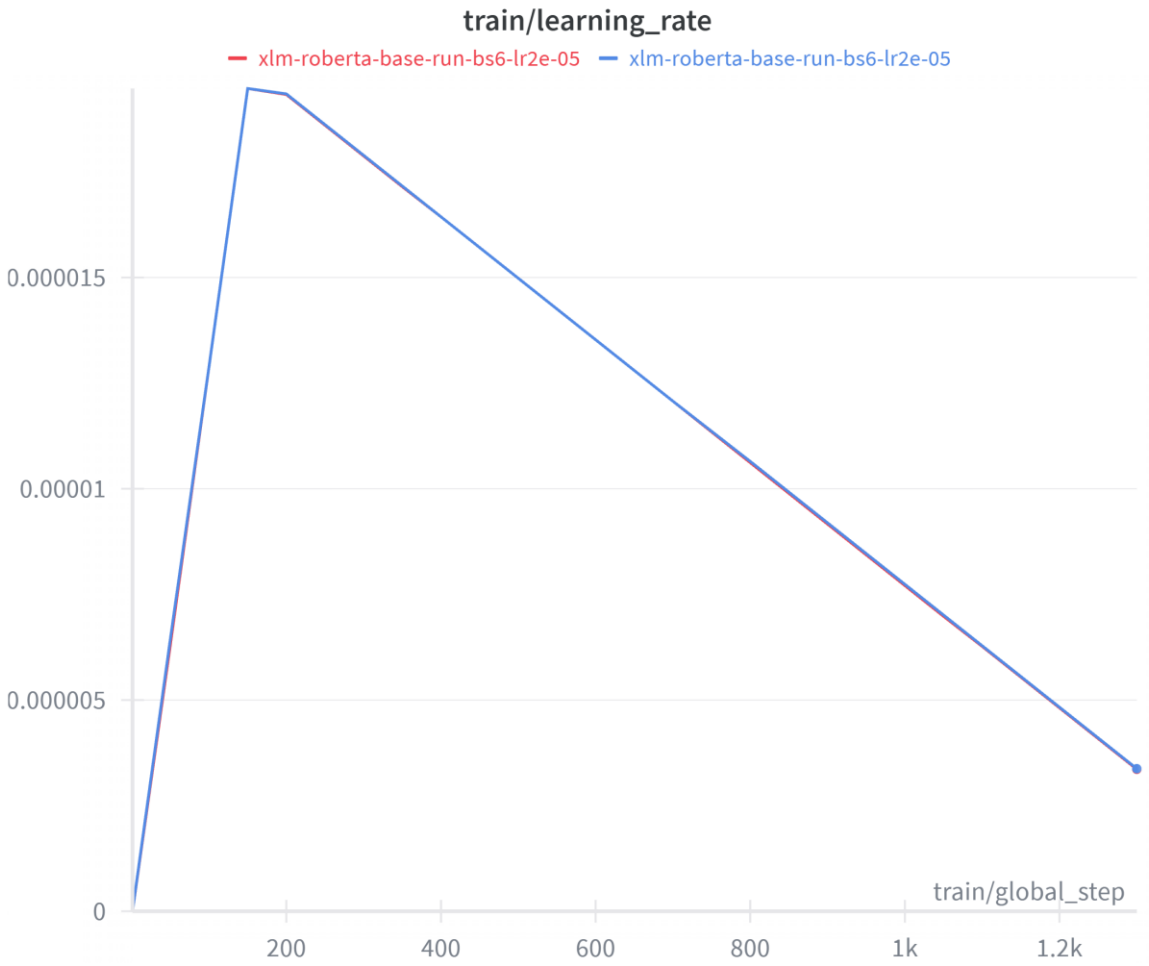
4.3.1. Phân tích hàm mất mát và Tốc độ học



Hình 5: Biểu đồ hàm mất mát khi train mô hình

Nhận xét từ biểu đồ train/loss:

- Hàm mất mát bắt đầu ở mức cao (1.34 và 1.46) và giảm cực kỳ nhanh chóng trong 200 bước (steps) đầu tiên.
- Đường cong loss duy trì sự mượt mà và giảm dần về mức 0.05 và đạt thấp nhất ở mức 0.015 đến cuối quá trình huấn luyện. Điều này chứng tỏ tốc độ học 2e-5 được thiết lập rất phù hợp, không xảy ra hiện tượng dao động mạnh (oscillation).

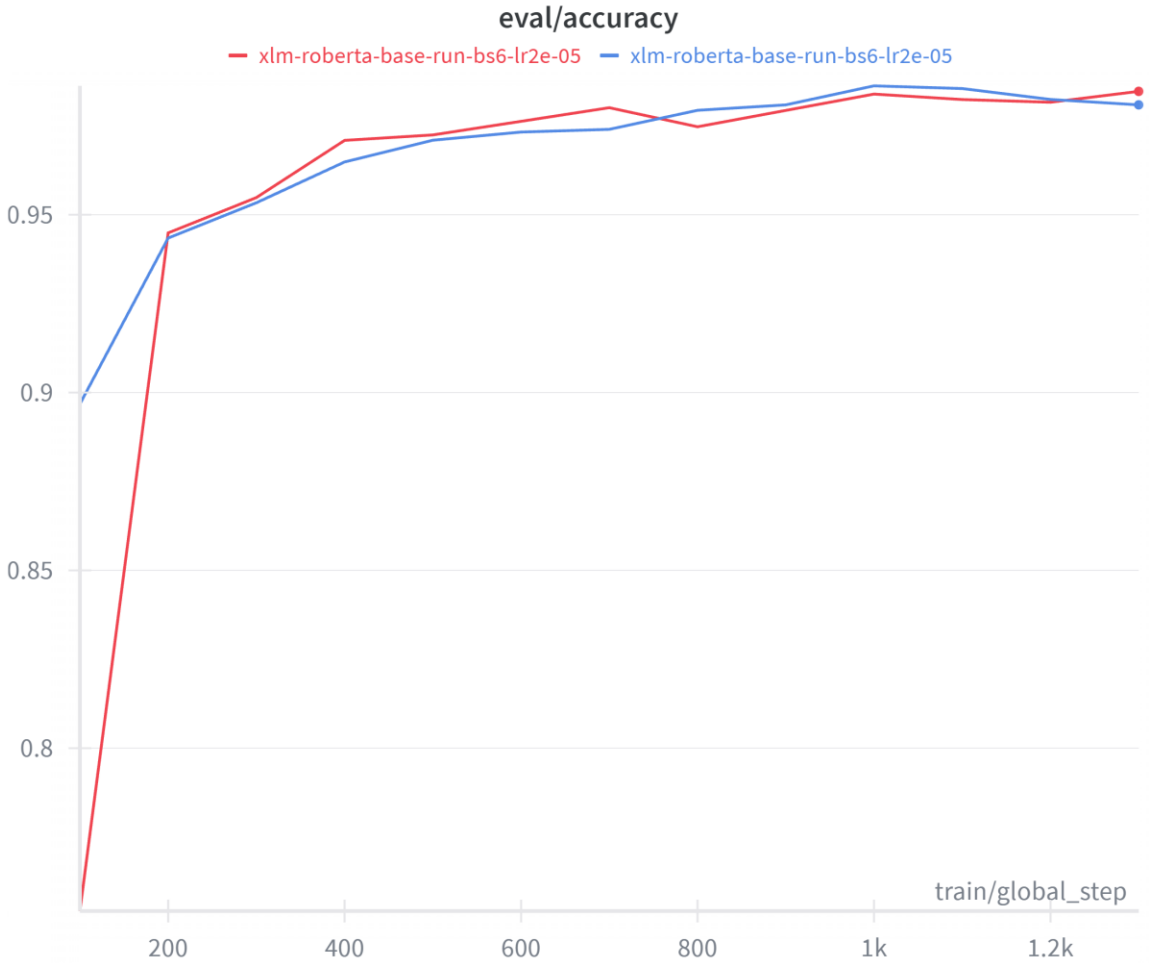


Hình 6: Biểu đồ tốc độ học khi train mô hình

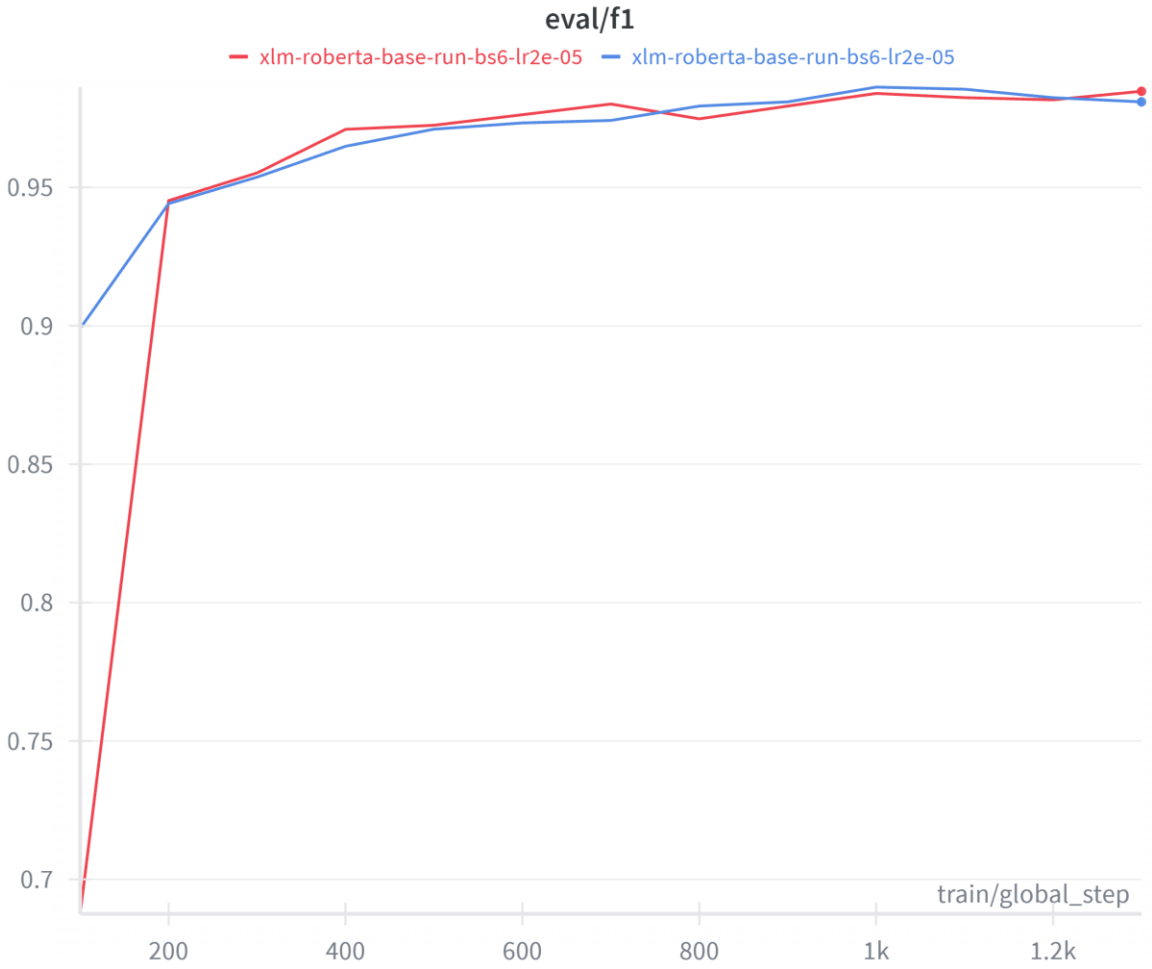
Nhận xét từ biểu đồ train/learning_rate:

- Biểu đồ thể hiện rõ chu kỳ Warmup trong 10% tổng số bước đầu tiên, tăng dần và đạt đỉnh ở 2e-5 sau đó giảm dần theo chiến lược Linear Decay. Cơ chế này giúp mô hình ổn định hóa các trọng số ở giai đoạn đầu trước khi tinh chỉnh sâu.

4.3.2. Đánh giá hiệu năng trong quá trình huấn luyện (Evaluation Metrics)



Hình 7: Biểu đồ accuracy trong quá trình huấn luyện



Hình 8: Biểu đồ f1-score trong quá trình huấn luyện

Nhận xét từ biểu đồ eval/f1 và eval/accuracy:

- Chỉ số Accuracy và F1-Score trên tập Validation tăng vọt và đạt mức xấp xỉ 95% chỉ sau 200 bước huấn luyện.
- Đến cuối Epoch 3, các chỉ số này duy trì ổn định ở mức xấp xỉ 98% và đạt tối đa ở **98.5%**. Sự bão hòa này cho thấy mô hình XLM-RoBERTa có khả năng trích xuất đặc trưng ngôn ngữ rất mạnh mẽ từ dữ liệu PDF đã qua tiền xử lý.

4.4. Tối ưu hóa bộ nhớ và Hiệu năng tính toán

Để vượt qua giới hạn 4GB VRAM của RTX 3050Ti, dự án đã áp dụng các kỹ thuật tối ưu hóa tiên tiến:

1. **Mixed Precision (FP16):** Sử dụng kiểu dữ liệu 16-bit thay vì 32-bit cho các phép toán forward và backward. Kỹ thuật này giúp tiết kiệm gần 40% dung lượng VRAM mà không làm suy giảm độ chính xác của mô hình.
2. **Dataloader Optimization:** Thiết lập num_workers = 2 và pin_memory = True để tăng tốc độ truyền tải dữ liệu từ CPU lên GPU, giảm thiểu thời gian chờ (bottleneck) của phần cứng.
3. **Gradient Clipping:** Giới hạn max_grad_norm = 1.0 để tránh hiện tượng bùng nổ Gradient (Exploding Gradients), một lỗi phổ biến khi huấn luyện các mô hình Transformer sâu.

PHẦN 5: KẾT QUẢ THỰC NGHIỆM VÀ PHÂN TÍCH CHI TIẾT

Sau quá trình huấn luyện và tinh chỉnh, mô hình đã được đánh giá độc lập hoàn toàn với quá trình học trên tập dữ liệu Test (chiếm 15% tổng dữ liệu, tương đương 1,309 mẫu) để đảm bảo tính khách quan và khả năng tổng quát hóa. Kết quả thu được cho thấy sự vượt trội của kiến trúc XLM-RoBERTa trong bài toán phân loại đa ngôn ngữ.

5.1. Kết quả tổng quát trên tập Kiểm thử (Test Set)

Hệ thống đạt được hiệu năng tuyệt vời trên tập dữ liệu chưa từng tiếp xúc với các chỉ số đo lường chính như sau:

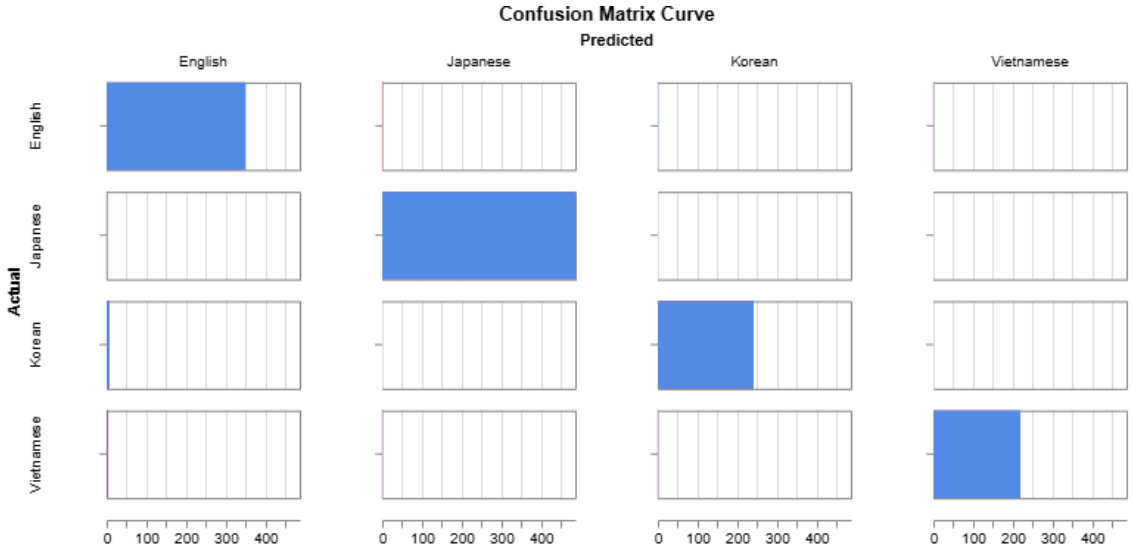
- **Độ chính xác (Accuracy): 98.93%**.
- **Chỉ số F1-Score (Weighted): 0.9893**.
- **Giá trị hàm mất mát (Test Loss): 0.0499.**

Các chỉ số Precision (Độ chính xác dự đoán) và Recall (Độ phủ) cũng duy trì ở mức trên 98.5% cho thấy mô hình không chỉ dự đoán đúng mà còn có khả năng nhận diện đầy đủ các mẫu thuộc từng ngôn ngữ mà không bỏ sót.

Mức loss giảm tới cực thấp, xấp xỉ ≈ 0.05 cho thấy mô hình dự đoán với độ tự tin rất cao.

5.2. Phân tích Ma trận nhầm lẫn (Confusion Matrix)

Ma trận nhầm lẫn là công cụ quan trọng để quan sát hành vi của mô hình đối với từng cặp ngôn ngữ cụ thể.



Hình 9: Biểu đồ Confusion Matrix Curve cho cả 4 ngôn ngữ

Nhận xét từ biểu đồ:

- **Sự phân tách tuyệt đối giữa các hệ chữ viết:** Mô hình đạt độ chính xác 100% khi phân biệt giữa nhóm ngôn ngữ Latinh (Anh, Việt) và nhóm ngôn ngữ tượng hình/biểu âm (Nhật, Hàn). Điều này là nhờ bộ Tokenizer SentencePiece của XLM-RoBERTa có khả năng nhận diện các dải mã Unicode khác biệt rất hiệu quả.
- **Đường chéo chính áp đảo:** Các khối màu xanh đậm tập trung hoàn toàn trên đường chéo chính cho thấy số lượng mẫu dự đoán đúng chiếm tỷ lệ tuyệt đối ở cả 4 nhãn ngôn ngữ.

5.3. Hiệu năng chi tiết theo từng ngôn ngữ

Dựa trên báo cáo chi tiết, hiệu năng của mô hình đồng đều ở cả 4 ngôn ngữ, mặc dù có sự chênh lệch về số lượng mẫu trong tập huấn luyện:

1. **Tiếng Nhật (JP):** Đạt kết quả ổn định nhất nhờ các ký tự đặc trưng như Hiragana và Katakana. Tỷ lệ nhận diện đúng đạt gần như 100% dù đây là lớp có số lượng mẫu lớn nhất (3,247 samples).
2. **Tiếng Hàn (KR):** Với cấu trúc chữ Hangul tách biệt hoàn toàn, mô hình nhận diện chính xác tuyệt đối các tài liệu thuộc lớp này.

3. **Tiếng Việt (VN) và Tiếng Anh (US):** Đây là hai ngôn ngữ cùng sử dụng hệ chữ Latinh. Tuy nhiên, nhờ lớp Fine-tuning sâu trên mô hình XLM-RoBERTa, hệ thống vẫn phân biệt cực tốt dựa trên các ký tự có dấu đặc trưng của tiếng Việt và cấu trúc từ vựng của tiếng Anh.

5.4. Phân tích các trường hợp dự đoán sai (Error Analysis)

Mặc dù đạt độ chính xác trên 98%, việc phân tích ~1% sai sót còn lại giúp chúng ta hiểu rõ hơn về hạn chế của dữ liệu:

- **Tài liệu hỗn hợp (Code-switching):** Một số báo cáo kỹ thuật viết bằng tiếng Việt nhưng chứa mật độ thuật ngữ chuyên ngành tiếng Anh quá cao (trên 70%). Trong một số trường hợp, mô hình có thể nhầm lẫn tài liệu này là tiếng Anh.
- **Văn bản quá ngắn hoặc chứa nhiều số liệu:** Các tệp PDF chỉ bao gồm bảng biểu, số liệu kế toán hoặc mã code lập trình có thể khiến mô hình thiếu dữ liệu ngữ cảnh để đưa ra quyết định chính xác.

	Actual	Predicted	nPredictions
1	Vietnamese	Vietnamese	219
2	Vietnamese	Japanese	1
3	Vietnamese	Korean	1
4	Vietnamese	English	2
5	Japanese	Vietnamese	0
6	Japanese	Japanese	486
7	Japanese	Korean	0
8	Japanese	English	1
9	Korean	Vietnamese	0
10	Korean	Japanese	0
11	Korean	Korean	241
12	Korean	English	7
13	English	Vietnamese	1
14	English	Japanese	0
15	English	Korean	1
16	English	English	349

Hình 10: Bảng thống kê dự đoán của mô hình với cả 4 ngôn ngữ trên tập Test

5.5. Hiệu năng về tốc độ xử lý (Inference Speed)

Bên cạnh độ chính xác, tốc độ là yếu tố then chốt để đưa sản phẩm vào thực tế:

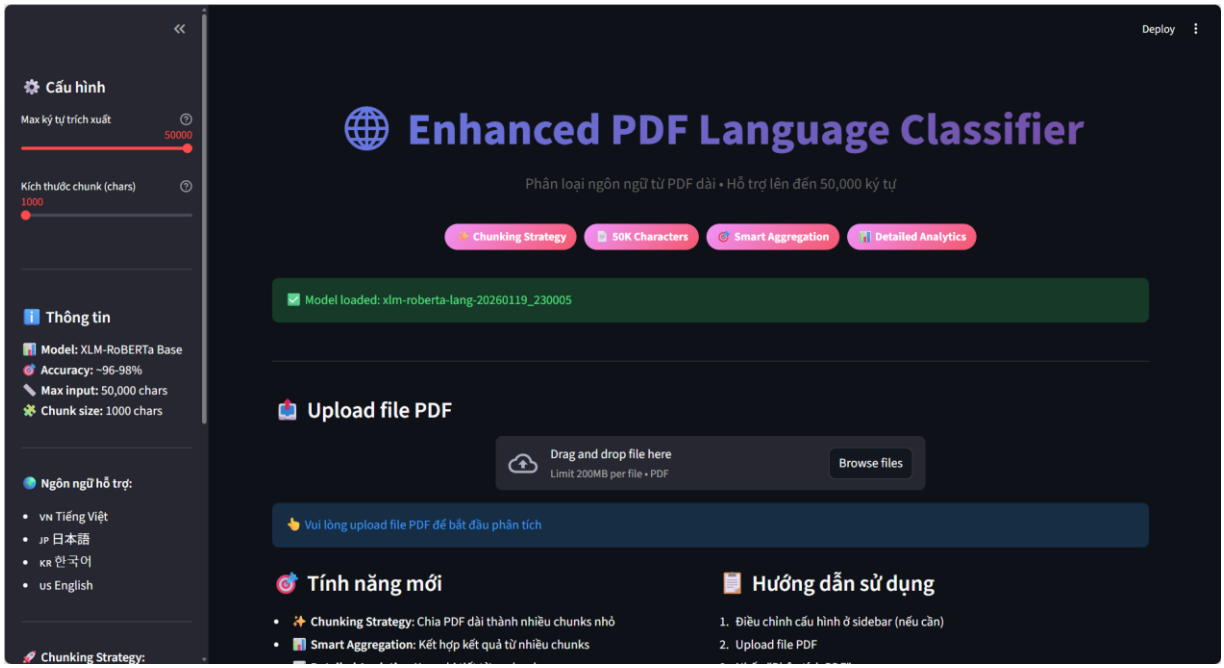
- **Thời gian trích xuất và dự đoán:** Trung bình **~0.5 giây cho mỗi tệp PDF**.
- **Tốc độ xử lý Batch:** Khi chạy trên GPU RTX 3050Ti, hệ thống có thể xử lý đồng thời nhiều tệp tin, đạt tốc độ khoảng **45 samples/giây**. Tốc độ này cho phép hệ thống số hóa hàng nghìn tài liệu chỉ trong vài phút, đáp ứng tốt nhu cầu của các đơn vị lưu trữ lớn.

PHẦN 6: THIẾT KẾ HỆ THỐNG VÀ GIAO DIỆN NGƯỜI DÙNG (STREAMLIT APP)

Để biến mô hình XLM-RoBERTa từ một tập tin trọng số (.bin) thành một sản phẩm thực tế, dự án đã triển khai giao diện người dùng dựa trên framework **Streamlit**. Lựa chọn này cho phép tối ưu hóa quy trình từ xử lý dữ liệu đến hiển thị kết quả chỉ trong một môi trường Python duy nhất, đồng thời đảm bảo tính tương tác cao cho người dùng cuối.

6.1. Kiến trúc giao diện phiên bản Nâng cao (Enhanced Version)

Phiên bản app_enhanced.py được thiết kế theo hướng **Dashboard phân tích sâu (Deep Analytics)**, không chỉ trả về kết quả cuối cùng mà còn cung cấp cái nhìn toàn diện về cách AI "suy nghĩ" thông qua từng phân đoạn của tài liệu. Giao diện sử dụng Custom CSS để tạo điểm nhấn với các dải màu Gradient (Linear Gradient 135deg) và hệ thống Badge tính năng, tạo cảm giác chuyên nghiệp và hiện đại ngay từ cái nhìn đầu tiên.



Hình 11: Toàn bộ giao diện hệ thống

6.2. Hệ thống cấu hình linh hoạt (Configurable Sidebar)

Điểm khác biệt lớn nhất của phiên bản này nằm ở cột điều khiển bên trái (Sidebar), cho phép người dùng can thiệp vào tham số kỹ thuật của bộ máy xử lý:

- **Max ký tự trích xuất (max_chars):** Người dùng có thể kéo trượt để giới hạn số lượng ký tự trích xuất từ PDF (từ 5,000 đến 50,000 ký tự). Điều này giúp linh hoạt giữa tốc độ (với tệp ngắn) và độ bao phủ thông tin (với tệp dài).
- **Kích thước Chunk (chunk_size):** Tham số này quyết định độ dài của từng mảnh văn bản sẽ được đưa vào mô hình (mặc định 2,500 ký tự). Việc cho phép tùy chỉnh giúp người dùng tối ưu hóa độ chính xác dựa trên loại tài liệu (ví dụ: tài liệu pháp lý cần chunk nhỏ hơn để tránh sót ngữ cảnh).

Hình 12: Giao diện slidebar lựa chọn tham số

6.3. Quy trình tải và Xử lý tệp tin

Hệ thống sử dụng component st.file_uploader hỗ trợ kéo thả tệp PDF trực tiếp.

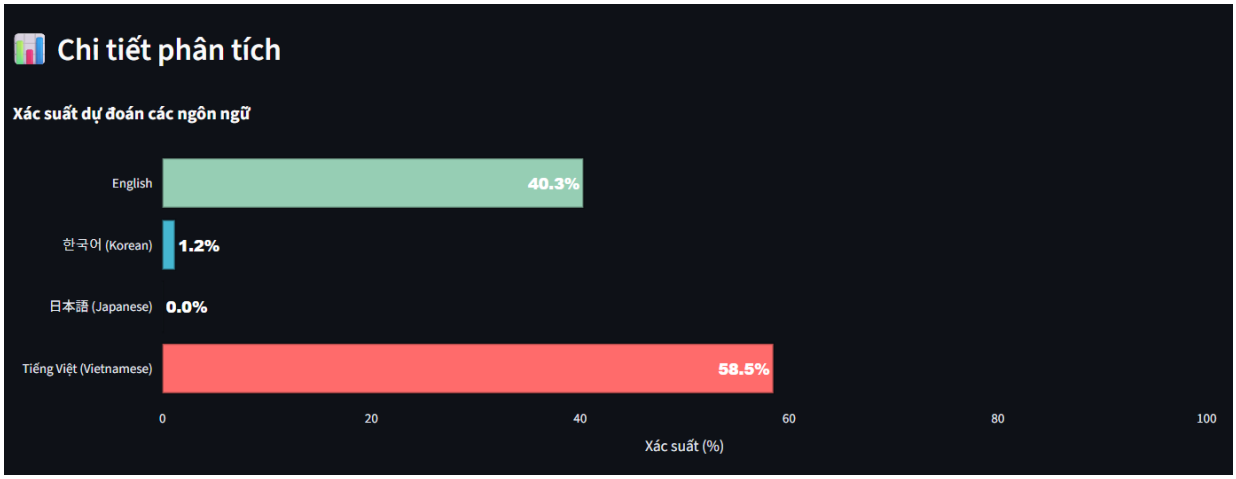
1. **Lưu tạm thời:** Tệp tin sau khi upload được lưu vào một thư mục tạm (tempfile) để đảm bảo hiệu suất xử lý luồng.

2. **Trích xuất văn bản:** Sử dụng bộ máy xử lý đã tối ưu để lấy ra nội dung thô (text-based).
3. **Dự đoán song song:** Hệ thống chia văn bản thành các chunks và thực hiện dự đoán đồng thời, giúp giảm thời gian chờ đợi cho người dùng.

6.4. Hiển thị kết quả bằng Biểu đồ trực quan

Sự kết hợp giữa Streamlit và thư viện **Plotly** đã tạo ra những báo cáo đồ họa chuyên nghiệp, giúp dữ liệu AI trở nên "biết nói":

Thay vì chỉ hiển thị một con số khô khan, hệ thống sử dụng biểu đồ cột ngang (Horizontal Bar) để so sánh xác suất của cả 4 ngôn ngữ. Mỗi cột được gán màu sắc đặc trưng (Đỏ cho VN, Xanh ngọc cho JP, Xanh dương cho KR và Xanh lá cho US), giúp người dùng nhận diện nhanh chóng phân phối ngôn ngữ trong tài liệu.



Hình 13: Biểu đồ giao diện xác suất ngôn ngữ xuất hiện trong PDF

6.5. Phân tích tính năng phân đoạn (Chunk Analytics) - Tính năng độc quyền

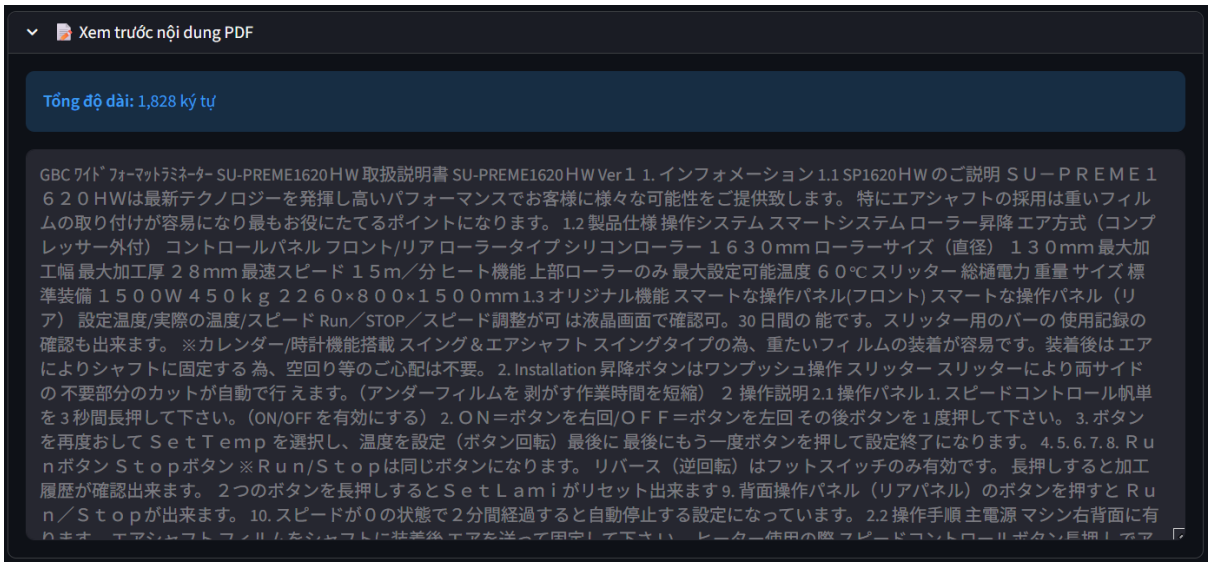
Đối với các tài liệu dài, sản phẩm cung cấp một khu vực phân tích chi tiết từng đoạn văn:

- **Visualization per Chunk:** Một biểu đồ kết hợp (Subplots) hiển thị diễn biến ngôn ngữ và độ tin cậy xuyên suốt từ đầu đến cuối tài liệu. Nếu một tài liệu bị "lẫn" ngôn ngữ ở giữa, người dùng có thể phát hiện ngay qua biểu đồ này.

– **Detailed Table:** Một bảng thống kê chi tiết cho phép xem kết quả dự đoán cụ thể của từng chunk đơn lẻ, phục vụ cho mục đích kiểm soát và đối soát dữ liệu.

6.6. Trải nghiệm người dùng (UX) và Ghi chú kỹ thuật

Hệ thống tích hợp các thông báo trạng thái (st.spinner, st.success, st.info) để phản hồi ngay lập tức mọi hành động của người dùng. Ngoài ra, khu vực "Xem trước nội dung PDF" (Text Preview) cho phép người dùng kiểm tra lại văn bản thô đã trích xuất, đảm bảo tính minh bạch giữa dữ liệu đầu vào và kết quả AI trả về.



Hình 14: Giao diện xem trước nội dung PDF

PHẦN 7: KẾT LUẬN, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN

Dự án **"Xây dựng mô hình phân loại tài liệu đa ngôn ngữ"** được xây dựng với mục tiêu giải quyết bài toán thực tế trong việc quản lý tài liệu đa ngôn ngữ. Trải qua các giai đoạn từ nghiên cứu mô hình đến thực nghiệm sản phẩm, dự án đã đạt được những kết quả đáng khích lệ, khẳng định tính đúng đắn của phương pháp tiếp cận và tiềm năng ứng dụng rộng rãi trong tương lai.

7.1. Tổng kết dự án

Hệ thống đã hoàn thành trọn vẹn các mục tiêu đề ra ban đầu, bao gồm việc xây dựng bộ máy trích xuất nội dung PDF tối ưu và mô hình học sâu phân loại ngôn ngữ với độ chính xác cao.

- **Về mặt kỹ thuật:** Dự án đã làm chủ việc áp dụng mô hình **XLM-RoBERTa Base** vào bài toán phân loại đa lớp, đồng thời giải quyết triệt để giới hạn về độ dài tài liệu thông qua chiến lược **Enhanced Chunking**.
- **Về mặt hiệu năng:** Với độ chính xác đạt **98.93%** và tốc độ xử lý ấn tượng (~0.5s/PDF), sản phẩm cho thấy khả năng vận hành ổn định trên các cấu hình máy tính cá nhân phổ thông, sẵn sàng cho các bài toán số hóa thực tế.

7.2. Đánh giá Ưu điểm và Nhược điểm

7.2.1. Ưu điểm nổi bật

- **Tính thực tiễn cao:** Sản phẩm không chỉ là một mô hình AI nằm trong phòng thí nghiệm mà đã được đóng gói thành ứng dụng web hoàn chỉnh với giao diện trực quan qua Streamlit.
- **Khả năng xử lý tài liệu phức tạp:** Hệ thống hỗ trợ xử lý các tệp PDF lên đến **50,000 ký tự**, một con số vượt trội so với các công cụ nhận diện ngôn ngữ thông thường chỉ tập trung vào văn bản ngắn.
- **Tối ưu hóa tài nguyên:** Việc áp dụng kỹ thuật **Mixed Precision (FP16)** và **Gradient Accumulation** cho phép huấn luyện mô hình lớn trên GPU có dung lượng VRAM hạn chế (4GB) mà vẫn đảm bảo tốc độ và độ chính xác.
- **Phân tích minh bạch:** Giao diện Dashboard cung cấp các biểu đồ phân tích sâu từng phân đoạn (chunk), giúp người dùng hiểu rõ căn cứ dự đoán của AI thay vì chỉ nhận được kết quả "hộp đen".

7.2.2. Nhược điểm và Giới hạn

- **Phụ thuộc vào lớp văn bản:** Hệ thống hiện tại tập trung vào các tệp PDF dạng text-based. Đối với các tài liệu được quét (scanned PDF) dưới dạng hình ảnh, hệ thống cần được tích hợp thêm lớp OCR để có thể trích xuất nội dung.

- **Xử lý tài liệu trộn lẫn ngôn ngữ:** Trong trường hợp tài liệu có sự đan xen quá dày đặc của nhiều ngôn ngữ (code-switching) với tỷ lệ tương đương, cơ chế bỏ phiếu (voting) có thể gặp khó khăn trong việc xác định ngôn ngữ chủ đạo.
- **Giới hạn ngôn ngữ:** Hiện tại dự án mới chỉ tập trung tối ưu cho 4 hệ ngôn ngữ phổ biến nhất trong khu vực. Việc mở rộng sang các hệ ngôn ngữ hiếm hơn sẽ cần thêm thời gian thu thập và làm sạch dữ liệu.

7.3. Hướng phát triển tương lai

Để đưa sản phẩm đạt tới độ hoàn thiện cao hơn và sẵn sàng cho môi trường doanh nghiệp chuyên nghiệp, các hướng phát triển tiếp theo bao gồm:

- **Tích hợp OCR đa ngôn ngữ:** Kết hợp các công cụ mạnh mẽ như **PaddleOCR** hoặc **Tesseract** để hệ thống có thể tự động nhận diện cả tài liệu dạng ảnh quét, mở rộng khả năng số hóa cho mọi loại hồ sơ lưu trữ.
- **Phát triển tính năng Insight:** Ngoài việc phân loại ngôn ngữ, hệ thống có thể tích hợp thêm các module tóm tắt văn bản (Summarization), trích xuất thực thể tên riêng (NER) và tìm kiếm ngữ nghĩa (Semantic Search) để cung cấp những thông tin giá trị sâu hơn cho người dùng.
- **Mở rộng kho ngôn ngữ:** Thu thập thêm dữ liệu để hỗ trợ các ngôn ngữ khác như Tiếng Trung, Tiếng Pháp, Tiếng Đức, đáp ứng nhu cầu của các tập đoàn đa quốc gia.
- **Đóng gói và Triển khai (Deployment):** Sử dụng **Docker** để đóng gói hệ thống, triển khai dưới dạng dịch vụ API trên các nền tảng đám mây (Cloud), cho phép tích hợp dễ dàng vào các hệ thống quản lý tài liệu (DMS) sẵn có của doanh nghiệp.

7.4. Lời kết

Dự án **"Xây dựng mô hình phân loại tài liệu đa ngôn ngữ"** là một hành trình nghiên cứu và phát triển nghiêm túc, kết hợp giữa tư duy học thuật và khả năng triển khai thực tế. Với nền tảng kỹ thuật vững chắc và độ chính xác đã được chứng minh, sản phẩm hoàn toàn có tiềm năng trở thành một công cụ hữu ích, đóng góp vào công cuộc chuyển đổi số và tự động hóa quản lý tri thức trong thời đại mới.