ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA TOÁN - TIN HỌC

BÁO CÁO

SEMINAR KHOA HỌC DỮ LIỆU

ĐỀ TÀI:

AUTOMATIC LABELLING ENGINE WITH LLM

SINH VIÊN THỰC HIỆN:

PHÙNG DŨNG QUÂN – MSSV: 22280073
NGUYỄN HỒ TUYÊN – MSSV: 22280103

GIẢNG VIÊN HƯỚNG DẪN:

ThS. ĐOÀN THỊ TRÂM

Mục lục

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

Lĩnh vực NLP và Machine Learning:

- ABSA - Aspect-Based Sentiment Analysis (Phân tích cảm xúc dựa trên khía cạnh)
- NLP - Natural Language Processing (Xử lý ngôn ngữ tự nhiên)
- LLM - Large Language Model (Mô hình ngôn ngữ lớn)
- BERT - Bidirectional Encoder Representations from Transformers
- RoBERTa - Robustly Optimized BERT Pretraining Approach
- DeBERTa - Decoding-enhanced BERT with Disentangled Attention
- DistilBERT - Distilled version of BERT
- LoRA - Low-Rank Adaptation
- QLoRA - Quantized Low-Rank Adaptation
- PEFT - Parameter-Efficient Fine-Tuning
- NSP - Next Sentence Prediction

Kỹ thuật và Định dạng:

- GGUF - GPT-Generated Unified Format
- NF4 - NormalFloat 4-bit
- FP16 - Floating Point 16-bit
- FP32 - Floating Point 32-bit
- LCS - Longest Common Subsequence
- JSON - JavaScript Object Notation
- CSV - Comma-Separated Values
- HTML - HyperText Markup Language
- CUDA - Compute Unified Device Architecture
- GPU - Graphics Processing Unit
- VRAM - Video Random Access Memory

Độ đo đánh giá:

- EM - Exact Match
- TP - True Positive
- FP - False Positive

- FN - False Negative
- ROUGE-L - Recall-Oriented Understudy for Gisting
- Evaluation - Longest Common Subsequence
- DQA - Data Quality Assurance

Danh sách bảng

TÓM TẮT ĐỒ ÁN

Sự phát triển của các nền tảng đặt phòng trực tuyến tạo ra khối lượng lớn dữ liệu đánh giá khách sạn, trong đó chứa nhiều thông tin quan trọng về trải nghiệm khách hàng theo từng khía cạnh dịch vụ. Tuy nhiên, việc phân tích và gán nhãn cảm xúc chi tiết cho các đánh giá này vẫn còn gặp nhiều khó khăn do dữ liệu không cấu trúc và cách diễn đạt đa dạng.

Nguyên nhân chủ yếu đến từ chi phí và thời gian lớn của quá trình gán nhãn thủ công, cũng như hạn chế của các phương pháp truyền thống khi xử lý câu phức, nhiều mệnh đề và ngữ nghĩa mơ hồ. Điều này đòi hỏi một hệ thống tự động có khả năng hiểu ngữ cảnh sâu và hoạt động hiệu quả trên tài nguyên tính toán hạn chế.

Đồ án đề xuất một Automatic Labelling Engine dựa trên kiến trúc pipeline lai, kết hợp mô hình ngôn ngữ lớn cho các tác vụ tách mệnh đề và trích xuất Aspect–Opinion, với các mô hình Transformer được tinh chỉnh bằng LoRA cho bài toán phân loại khía cạnh và cảm xúc. Hệ thống tự động hóa toàn bộ quy trình từ xử lý dữ liệu đến xuất kết quả gán nhãn có cấu trúc.

Trong tương lai, hệ thống có thể được mở rộng bằng cách tăng cường dữ liệu cho các nhãn hiếm, áp dụng các kỹ thuật xử lý mất cân bằng, tinh chỉnh mô hình ngôn ngữ lớn theo miền khách sạn và tích hợp cơ chế học liên tục từ phản hồi người dùng, hướng tới triển bảo triển khai trong các hệ thống phân tích và dashboard thời gian thực.

I. Giới thiệu

1. Lý do chọn đề tài

Sự bùng nổ của các nền tảng đặt phòng trực tuyến như Booking.com, Agoda, Traveloka,… tạo ra khối lượng lớn review khách sạn mỗi ngày, chứa nhiều thông tin quan trọng về trải nghiệm khách hàng nhưng rất khó phân tích thủ công do số lượng lớn và cách diễn đạt đa dạng. Trên thực tế, nhiều đơn vị chỉ dừng ở việc theo dõi điểm đánh giá trung bình mà chưa khai thác sâu cảm xúc của khách theo từng khía cạnh cụ thể như phòng, nhân viên, vị trí, vệ sinh, tiện ích hay giá cả, dẫn đến hạn chế trong việc nhận diện chính xác điểm mạnh và điểm yếu của dịch vụ. Sự phát triển của các kỹ thuật xử lý ngôn ngữ tự nhiên và mô hình ngôn ngữ lớn cho phép phân tích review ở mức chi tiết hơn, vì vậy việc nghiên cứu và xây dựng một Automatic Labelling Engine có khả năng tự động gán nhãn khía cạnh và cảm xúc cho review khách sạn là hết sức cần thiết và mang lại giá trị ứng dụng cao cho ngành khách sạn.

2. Mục tiêu của đề tài

Trong nghiên cứu này, chúng tôi phát triển một hệ thống gán nhãn tự động dựa trên mô hình ngôn ngữ lớn (LLMs) để xử lý dữ liệu đánh giá khách sạn mà không cần can thiệp thủ công. Hệ thống khai thác khả năng hiểu ngữ cảnh, suy luận và phân loại của LLMs nhằm tạo nhãn chính xác về cảm xúc, mức độ hài lòng và đặc điểm dịch vụ. Mục tiêu chính:

- Giảm chi phí và thời gian gán nhãn thủ công thông qua tự động hóa.
- Tăng độ chính xác, nhất quán, giảm sai lệch con người.
- Tự động hóa toàn bộ quy trình từ tiền xử lý đến xuất nhãn, hỗ trợ mô hình downstream như phân loại cảm xúc và phát hiện vấn đề dịch vụ.
- Đảm bảo khả năng mở rộng và tích hợp linh hoạt vào quy trình học máy cho doanh nghiệp khách sạn hoặc nghiên cứu.

Kiến trúc mô-đun kết hợp prompting, few-shot learning, rule augmentation và active learning để tối ưu hóa hiệu suất trên dữ liệu văn bản đa dạng.

Hệ thống được kỳ vọng cung cấp cơ chế gán nhãn tự động hiệu quả, đóng vai trò làm nguồn dữ liệu đầu vào cho các bài toán phân tích và học máy trong lĩnh vực khách sạn.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu bao gồm:

- Dữ liệu review khách sạn dạng văn bản do khách hàng để lại trên các nền tảng đặt phòng trực tuyến.
- Bài toán tự động gán nhãn khía cạnh dịch vụ và cảm xúc tương ứng (Aspect-Based Sentiment Analysis).

Phạm vi nghiên cứu được giới hạn như sau:

- Chỉ xem xét và xử lý review ở dạng văn bản, không bao gồm hình ảnh, video hay dữ liệu âm thanh.
- Tập trung vào một số nhóm khía cạnh cốt lõi: Experience, Amenity, Service, Facility, Branding, Loyalty.
- Cảm xúc được phân loại ở ba mức: tích cực, tiêu cực và trung tính.
- Hệ thống được thiết kế, xây dựng và đánh giá trong môi trường thử nghiệm, chưa hướng tới triển khai ở quy mô sản phẩm thương mại.

II. Cơ sở lý thuyết

1. Tổng quan về Phân tích cảm xúc dựa trên khía cạnh (ABSA)

1.1. Định nghĩa và Mục tiêu

Aspect-Based Sentiment Analysis (ABSA) là một nhánh chuyên sâu của xử lý ngôn ngữ tự nhiên (NLP), tập trung vào việc xác định cảm xúc của văn bản đối với từng khía cạnh (aspect) hoặc thực thể cụ thể thay vì chỉ gán một nhãn cảm xúc chung cho toàn bộ văn bản. Mục tiêu cốt lõi của ABSA trong hệ thống là giải quyết hai vấn đề chính:

1. **Aspect Extraction:** Nhận diện các khía cạnh được nhắc đến (ví dụ: "phòng ốc", "nhân viên").
2. **Sentiment Classification:** Xác định thái độ (Tích cực, Tiêu cực, Trung tính) đối với từng khía cạnh đó.

1.2. Các thành phần trong bài toán ABSA

Một hệ thống ABSA tiêu chuẩn gồm bốn thành phần chính:

- **Aspect Term:** Từ ngữ chỉ khía cạnh cụ thể xuất hiện trong câu văn (ví dụ: *"receptionist"*, *"AC"*).
- **Opinion Term:** Từ ngữ thể hiện quan điểm (ví dụ: *"helpful"*, *"broken"*).
- **Aspect Category:** Nhóm chủ đề được định nghĩa (ví dụ: *Service, Facility, Amenity, Experience, Loyalty, Branding*).
- **Sentiment Polarity:** Chiều hướng cảm xúc (*Positive, Negative, Neutral*).

1.3. Ưu điểm và Hạn chế của ABSA

Ưu điểm:

- *Độ chi tiết cao (Fine-grained):* Cho phép phân tích đa chiều. Ví dụ: một câu review có thể khen "vị trí" nhưng chê "giá cả", điều mà phân tích cảm xúc truyền thống không tách biệt được.
- *Hỗ trợ ra quyết định:* Cung cấp thông tin cụ thể để doanh nghiệp cải thiện từng mảng dịch vụ.

Hạn chế và Thách thức:

- *Độ phức tạp ngữ nghĩa:* Khó xử lý các câu có cấu trúc phức tạp, mỉa mai (sarcasm) hoặc các khía cạnh ẩn (implicit aspects) không được nhắc tên trực tiếp.
- *Phụ thuộc dữ liệu gán nhãn:* Yêu cầu lượng lớn dữ liệu được gán nhãn chi tiết ở mức độ khía cạnh, gây tốn kém chi phí xây dựng.

2. Các mô hình Ngôn ngữ được sử dụng

Hệ thống sử dụng kiến trúc lai (Hybrid) kết hợp giữa Mô hình ngôn ngữ lớn (Generative LLMs) và Mô hình ngôn ngữ mã hóa (Encoder-based Models).

2.1. Mô hình ngôn ngữ lớn (LLM) và Qwen/Phi

Trong các tác vụ trích xuất thông tin (Extraction), hệ thống sử dụng các mô hình ngôn ngữ lớn có khả năng sinh văn bản (Generative AI). Các mô hình này (như Qwen hoặc Phi) hoạt động dựa trên kiến trúc Decoder-only của Transformer, được huấn luyện trên lượng dữ liệu khổng lồ để hiểu và sinh ngữ cảnh.

- **Vai trò trong hệ thống:** LLM đóng vai trò là bộ xử lý lý luận (reasoning engine), thực hiện tách câu (Clause split) và trích xuất các cặp thuật ngữ (Term extraction) dựa trên prompt hướng dẫn.
- **Instruction Tuning:** Mô hình được tinh chỉnh để tuân thủ các chỉ dẫn cụ thể (instruction-following), giúp đầu ra tuân thủ định dạng JSON hoặc cấu trúc quy định.

2.2. RoBERTa (Robustly Optimized BERT Pretraining Approach)

RoBERTa là một biến thể cải tiến của BERT, sử dụng kiến trúc Encoder-only. RoBERTa loại bỏ nhiệm vụ dự đoán câu tiếp theo (NSP) của BERT và sử dụng cơ chế mặt nạ động (dynamic masking) với lượng dữ liệu huấn luyện lớn hơn.

- **Đặc điểm:** RoBERTa vượt trội trong các tác vụ hiểu ngôn ngữ và phân loại văn bản nhờ khả năng biểu diễn ngữ cảnh hai chiều (bidirectional context).
- **Ứng dụng:** Trong hệ thống, RoBERTa được sử dụng cho bài toán *Category Classification* (phân loại khía cạnh) nhờ độ chính xác cao trong việc phân biệt các lớp ngữ nghĩa.

2.3. DeBERTa/DistilBERT

Đây là các mô hình Transformer được tối ưu hóa về hiệu năng hoặc kiến trúc.

- **Ứng dụng:** Được sử dụng cho bài toán *Polarity Classification* (phân loại cảm xúc) để xác định nhãn Positive/Negative/Neutral cho từng mệnh đề.

3. Các kỹ thuật tối ưu hóa mô hình

Để vận hành các mô hình ngôn ngữ phức tạp trên tài nguyên phần cứng giới hạn, đề tài áp dụng các kỹ thuật tối ưu hóa tiên tiến.

3.1. Quantization

Lượng tử hóa là kỹ thuật giảm độ chính xác của các trọng số mô hình từ dấu chấm động 32-bit (FP32) hoặc 16-bit (FP16) xuống các định dạng thấp hơn như 4-bit hoặc 8-bit mà không làm giảm đáng kể độ chính xác của mô hình.

- **Định dạng GGUF/NF4:** Hệ thống sử dụng định dạng GGUF (cho Qwen) và NF4 (NormalFloat 4-bit cho RoBERTa) để giảm dung lượng bộ nhớ VRAM, cho phép chạy mô hình LLM cục bộ trên GPU phổ thông.

3.2. LoRA - Low-Rank Adaptation

LoRA (Low-Rank Adaptation) là kỹ thuật Parameter-Efficient Fine-Tuning (PEFT). Thay vì cập nhật toàn bộ trọng số của mô hình khi huấn luyện lại (fine-tuning), LoRA đóng băng các trọng số mô hình gốc và chỉ huấn luyện các ma trận phân rã hạng thấp (low-rank matrices) được chèn vào các lớp Attention.

- **Lợi ích:** Giảm số lượng tham số cần huấn luyện xuống dưới 1% so với mô hình gốc, giúp tiết kiệm thời gian và tài nguyên tính toán trong khi vẫn đạt hiệu quả cao trên dữ liệu.

4. Các độ đo đánh giá hệ thống

Để đánh giá hiệu quả của hệ thống, đề tài chia bài toán thành hai nhóm chính: (i) bài toán có đầu ra rời rạc và (ii) bài toán có đầu ra không rời rạc. Tương ứng với mỗi nhóm, các độ đo đánh giá được lựa chọn sao cho phù hợp với đặc thù của bài toán.

4.1. Generative outputs

Đối với các bài toán mà kết quả dự đoán có dạng nhãn hoặc chuỗi rời rạc, đề tài sử dụng các độ đo sau:

- **Exact Match (EM):** Đánh giá mức độ trùng khớp hoàn toàn giữa kết quả dự đoán và nhãn tham chiếu. Dự đoán chỉ được coi là đúng khi hai chuỗi giống nhau hoàn toàn.
- **F1-score mức token (Token-level F1-score):** Đo lường mức độ chồng lấp giữa các token trong kết quả dự đoán và nhãn tham chiếu, thông qua Macro Average của Precision và Recall ở mức token.
- **ROUGE-L:** Dựa trên độ dài chuỗi con chung dài nhất (Longest Common Subsequence – LCS) giữa kết quả dự đoán và văn bản tham chiếu, thường được sử dụng trong các bài toán sinh văn bản.
- **Embedding Similarity:** Đo độ tương đồng ngữ nghĩa giữa kết quả dự đoán và nhãn tham chiếu bằng cosine similarity của vector embedding, được sinh bởi mô hình SentenceTransformer ('all-MiniLM-L6-v2').

4.2. Discrete classification outputs

Đối với các bài toán phân loại truyền thống với đầu ra không rời rạc, đề tài sử dụng các độ đo tiêu chuẩn trong khai phá dữ liệu, bao gồm:

- **Accuracy (Độ chính xác):** Tỷ lệ số mẫu được dự đoán đúng trên tổng số mẫu.
- **Precision (Độ chính xác dự báo):** Tỷ lệ số mẫu dương tính được dự đoán đúng trên tổng số mẫu được dự đoán là dương tính.
- **Recall (Độ phủ):** Tỷ lệ số mẫu dương tính được dự đoán đúng trên tổng số mẫu dương tính thực tế.
- **F1-Score:** Trung bình của Precision và Recall, thường được sử dụng khi dữ liệu mất cân bằng nhằm đánh giá tổng quát hiệu năng mô hình.

III. Phương pháp thực hiện

1. Phương pháp nghiên cứu

Đề tài áp dụng phương pháp tiếp cận lai (Hybrid Approach), kết hợp sức mạnh suy luận của các Mô hình ngôn ngữ lớn (Large Language Models - LLMs) với sự chuyên biệt hóa của các mô hình học sâu truyền thống (Encoder-based Models). Quy trình nghiên cứu được thực hiện qua các bước:

1. **Thu thập và Xây dựng dữ liệu:** Sử dụng kỹ thuật cào dữ liệu (Web Crawling) để thu thập mẫu thực tế, sau đó tiến hành gán nhãn thủ công tuân theo hướng dẫn (guideline) nghiêm ngặt để tạo tập dữ liệu chuẩn (Ground Truth).

2. **Phát triển Pipeline xử lý:** Xây dựng hệ thống theo kiến trúc đường ống tuần tự, trong đó nhiệm vụ phức tạp (hiểu ngữ nghĩa, tách câu) được giao cho LLM, còn nhiệm vụ phân loại (Category, Polarity) được giao cho các mô hình nhỏ hơn đã được tinh chỉnh (Fine-tuning).
3. **Thực nghiệm và Đánh giá:** Huấn luyện mô hình trên tập dữ liệu đã gán nhãn và đánh giá hiệu quả thông qua các chỉ số như Accuracy, F1-Score.

2. Thu thập và Xử lý dữ liệu

2.1. Phương pháp thu thập dữ liệu

Dữ liệu đầu vào của hệ thống là các đánh giá (reviews) khách sạn được thu thập từ các nền tảng đặt phòng trực tuyến phổ biến như Booking.com, Agoda, Traveloka.

Quy trình thu thập dữ liệu tự động được thực hiện thông qua các thư viện lập trình Python chuyên dụng:

- **Selenium:** Được sử dụng để giả lập trình duyệt, xử lý các trang web động (dynamic content) yêu cầu cuộn trang hoặc tương tác JavaScript để tải thêm bình luận.
- **BeautifulSoup:** Được sử dụng để phân tích cú pháp HTML (parsing), trích xuất nội dung văn bản sạch từ mã nguồn trang web sau khi đã được tải về.

Dữ liệu thô sau khi thu thập sẽ trải qua bước tiền xử lý để loại bỏ các ký tự đặc biệt, chuẩn hóa mã hóa (Unicode) và loại bỏ các bình luận rác hoặc quá ngắn trước khi đưa vào gán nhãn.

2.2. Định nghĩa các khái niệm và đơn vị dữ liệu

Hệ thống xử lý dữ liệu dựa trên các đơn vị thông tin được định nghĩa cụ thể như sau:

- **Mệnh đề (Clause):** Là đơn vị ngữ nghĩa nhỏ nhất mà hệ thống xử lý. Một câu bình luận dài có thể bao gồm nhiều mệnh đề ghép lại. Việc tách nhỏ thành clause giúp cô lập ý kiến, tránh trường hợp một câu chứa nhiều cảm xúc trái ngược nhau.
- **Khía cạnh (Term) và Ý kiến (Opinion):**
  - *Term:* Là từ hoặc cụm từ chỉ đối tượng cụ thể được nhắc đến (ví dụ: "giường", "lễ tân").
  - *Opinion:* Là từ ngữ thể hiện tính chất hoặc cảm xúc gắn liền với Term đó (ví dụ: "êm ái", "nhiệt tình").
- **Chủ đề (Category):** Mỗi mệnh đề được phân loại vào một trong 6 nhóm chủ đề định sẵn, bao gồm:

Bảng 1: Danh mục các khía cạnh trong lĩnh vực khách sạn

Khía cạnh	Mô tả
Cơ sở vật chất (Facility)	Bao gồm các yếu tố như thiết bị, nội thất phòng, trang trí khách sạn, thiết kế nội thất, ban công và khu vực hồ bơi.
Tiện ích (Amenity)	Bao gồm các dịch vụ công cộng như bãi đậu xe, spa, nhà hàng, quà lưu niệm, các điểm đến lân cận, các tùy chọn thanh toán và sự an ninh.
Dịch vụ (Service)	Liên quan đến chất lượng dịch vụ của khách sạn, chất lượng món ăn và thái độ của nhân viên.
Trải nghiệm (Experience)	Liên quan đến trải nghiệm của khách hàng về khách sạn như bầu không khí và cảm nhận về mức độ thư giãn mà khách sạn mang lại.
Thương hiệu (Branding)	Phản ánh mức độ hài lòng chung của khách hàng so với thương hiệu được cung cấp.
Mức độ trung thành (Loyalty)	Cho biết khả năng khách hàng quay lại và giới thiệu khách sạn cho người khác.

- **Cảm xúc phân cực (Polarity Sentiment):** Là nhãn cảm xúc cuối cùng được gán cho mệnh đề, thuộc một trong ba giá trị:

Bảng 2: Các nhóm phân cực cảm xúc

Nhóm cảm xúc	Mô tả
Tích cực (Positive)	Bao gồm các cảm xúc như hài lòng, vui vẻ và hạnh phúc.
Tiêu cực (Negative)	Bao gồm các cảm xúc như không hài lòng, thất vọng, phàn nàn và khó chịu.
Trung tính (Neutral)	Bao gồm các cảm xúc không mang ý nghĩa tích cực hoặc tiêu cực, cảm xúc bình thường và không có cảm xúc đặc biệt.

3. Yêu cầu hệ thống

Để đảm bảo tính ứng dụng thực tế, hệ thống *Automatic Labelling Engine* được xây dựng dựa trên các yêu cầu sau:

Yêu cầu chức năng:

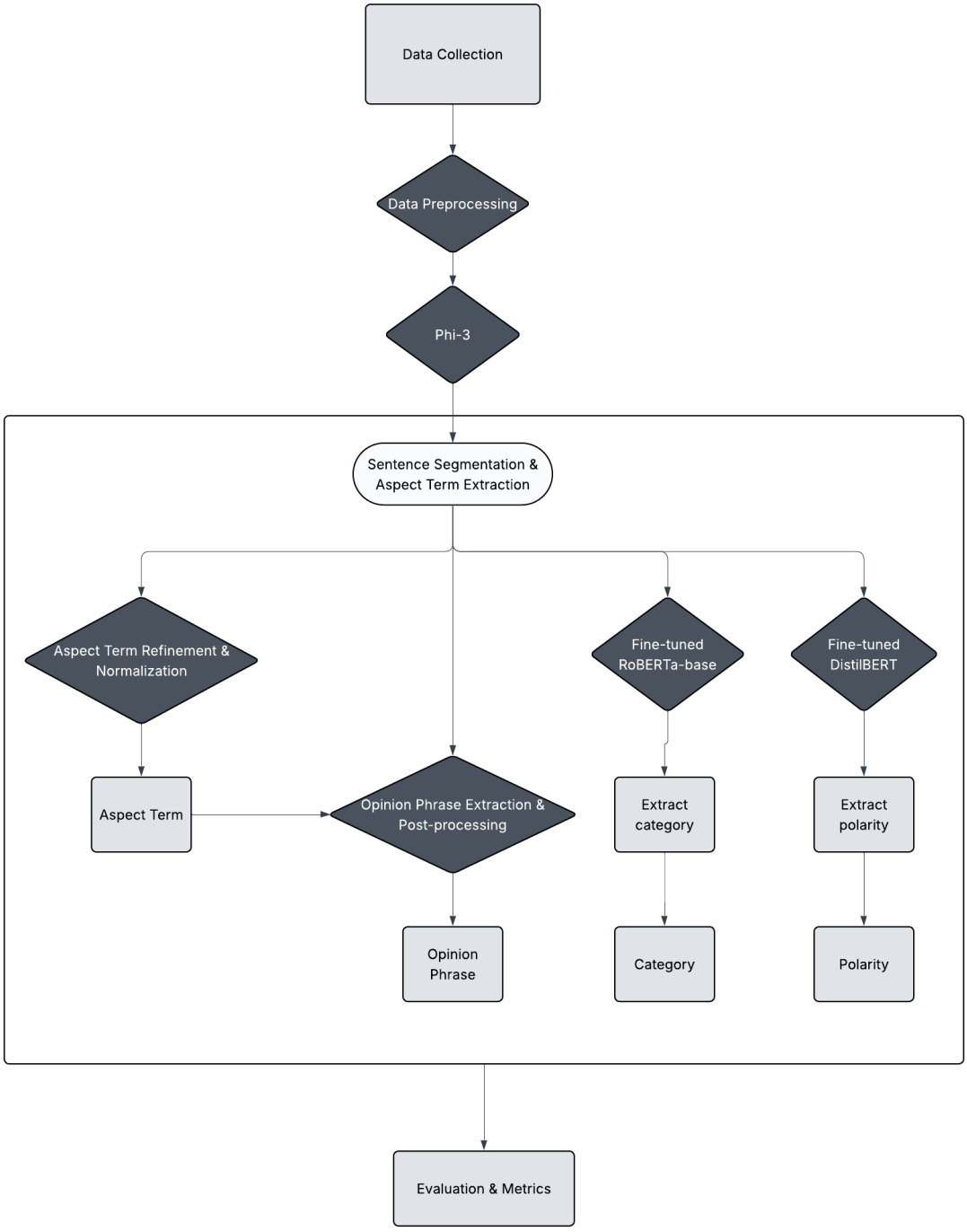
- Hệ thống phải nhận đầu vào là văn bản tự do (text review) từ file (CSV, Excel, TXT) hoặc nhập trực tiếp.
- Tự động thực hiện toàn bộ quy trình: Tách câu → Trích xuất Term, Opinion → Phân loại Category → Phân loại Polarity.
- Kết quả đầu ra phải có cấu trúc (JSON/DataFrame) để phục vụ lưu trữ và trực quan hóa (Dashboard).

Yêu cầu phi chức năng:

- *Độ chính xác:* Đạt mức tin cậy cao trên các độ đo F1-Score, đặc biệt là khả năng phân biệt các khía cạnh dễ nhầm lẫn giữa các cặp (Facility và Amenity), (Branding và Experience).
- *Hiệu năng:* Thời gian phản hồi chấp nhận được, tối ưu hóa việc sử dụng tài nguyên thông qua kỹ thuật Batch Processing và Quantization (4-bit) để chạy được trên phần cứng giới hạn.

- *Khả năng mở rộng:* Kiến trúc phải cho phép thay thế mô hình (ví dụ: đổi từ Qwen sang Phi-3) mà không phá vỡ luồng xử lý chung.

4. Kiến trúc tổng thể hệ thống



Hình 1: Kiến trúc tổng thể của hệ thống phân tích và gán nhãn đánh giá dựa trên phương pháp pipeline lai

Hình 1 minh họa kiến trúc tổng thể của hệ thống được đề xuất. Hệ thống được thiết kế theo kiến trúc pipeline lai (*Hybrid Pipeline Architecture*), kết hợp giữa Mô hình Ngôn ngữ Lớn (*Large Language Models – LLMs*) và các mô hình học sâu encoder-base đã được tinh chỉnh.

Cách tiếp cận này nhằm tận dụng khả năng suy luận ngữ nghĩa mạnh mẽ của LLM cho các tác vụ phức tạp, đồng thời khai thác tính hiệu quả và ổn định của các mô hình nhỏ hơn cho các nhiệm vụ phân

loại chuyên biệt.

Luồng xử lý của hệ thống được tổ chức theo hướng tuần tự, trong đó đầu ra của mỗi mô-đun đóng vai trò là đầu vào cho mô-đun kế tiếp.

Khối thu thập dữ liệu (Data Collection)

Khối thu thập dữ liệu chịu trách nhiệm thu thập dữ liệu đầu vào từ các nguồn thực tế thông qua kỹ thuật *Web Crawling*. Dữ liệu chủ yếu là các văn bản đánh giá (review) của người dùng, mang tính không cấu trúc, đa dạng về độ dài, ngôn ngữ và cách biểu đạt.

Đây là nguồn dữ liệu thô ban đầu, làm cơ sở cho toàn bộ quá trình xử lý và phân tích của hệ thống.

Khối tiền xử lý dữ liệu (Data Preprocessing)

Dữ liệu thô sau khi thu thập được đưa vào khối tiền xử lý nhằm làm sạch và chuẩn hóa văn bản.

Các bước tiền xử lý chính bao gồm:

- Loại bỏ các ký tự đặc biệt và thành phần nhiễu không cần thiết;
- Chuẩn hóa chữ hoa và chữ thường;
- Chuẩn hóa khoảng trắng và định dạng văn bản.

Khối tiền xử lý giúp cải thiện chất lượng dữ liệu đầu vào, từ đó giảm thiểu ảnh hưởng của nhiễu đến các bước xử lý ngữ nghĩa và huấn luyện mô hình phía sau.

Khối suy luận ngữ nghĩa bằng LLM

Sau khi tiền xử lý, dữ liệu được đưa vào mô hình ngôn ngữ lớn, đóng vai trò trung tâm trong kiến trúc hệ thống.

Khối LLM đảm nhiệm các tác vụ yêu cầu khả năng hiểu ngữ nghĩa sâu và suy luận theo ngữ cảnh, bao gồm:

- Phân tích ngữ nghĩa tổng thể của văn bản;
- Tách câu dựa trên ngữ cảnh (*Semantic Sentence Segmentation*);
- Trích xuất các *Aspect Term* tiềm năng từ mỗi câu hoặc mệnh đề.

Việc sử dụng LLM ở giai đoạn này cho phép hệ thống xử lý hiệu quả các câu phức tạp, đa ý, vốn là thách thức lớn đối với các phương pháp truyền thống.

Nhánh 1: Trích xuất và xử lý Aspect Term – Opinion

Từ kết quả suy luận của mô hình ngôn ngữ lớn (LLM), văn bản đánh giá ban đầu được chia thành các câu hoặc mệnh đề (*clause*) mang ý nghĩa ngữ nghĩa độc lập.

Quá trình này được thực hiện thông qua kỹ thuật *Prompt Engineering*, trong đó LLM được hướng dẫn rõ ràng để thực hiện đồng thời hai nhiệm vụ: (i) tách câu dựa trên ngữ cảnh và (ii) trích xuất các *Aspect Term* tiềm năng xuất hiện trong từng mệnh đề.

Kết quả đầu ra của giai đoạn này là các cặp dữ liệu có cấu trúc, bao gồm *Clause* và tập *Aspect Term* tương ứng, làm đầu vào cho các mô-đun xử lý chuyên biệt phía sau.

Khối tinh chỉnh và chuẩn hóa Aspect Term Các *Aspect Term* được trích xuất từ LLM có thể tồn tại dưới nhiều biến thể bao gồm cách diễn đạt không chuẩn. Do đó, khối này thực hiện quá trình tinh chỉnh và chuẩn hóa đầu ra thành một dạng để dễ dàng trích xuất được các label cần thiết.

Quá trình này giúp đảm bảo tính nhất quán của dữ liệu và nâng cao độ chính xác cho các bước phân tích tiếp theo.

Khối trích xuất Opinion Phrase Dựa trên các mệnh đề đã được tách và tập Aspect Term đã được chuẩn hóa, hệ thống tiếp tục sử dụng LLM với chiến lược *Prompt Engineering* để trích xuất các *Opinion Phrase*.

Trong đó, LLM được hướng dẫn tập trung vào việc xác định các cụm từ thể hiện quan điểm, đánh giá hoặc cảm xúc của người dùng đối với từng Aspect Term cụ thể, thay vì chỉ trích xuất các từ cảm xúc đơn lẻ.

Các bước hậu xử lý được áp dụng nhằm loại bỏ nhiễu, chuẩn hóa biểu đạt ngôn ngữ và đảm bảo mối liên kết rõ ràng giữa *Aspect Term* và *Opinion Phrase*.

Nhánh 2: Phân loại danh mục khía cạnh (Aspect Category Classification)

Nhánh phân loại danh mục khía cạnh có nhiệm vụ xác định loại khía cạnh (*Aspect Category*) mà mỗi mệnh đề (*Clause*) đề cập đến.

Việc thực hiện phân loại ở mức Clause giúp mô hình khai thác đầy đủ ngữ cảnh cục bộ, đồng thời giảm nhiễu so với việc phân loại trên các cụm từ rời rạc.

Đầu vào của mô hình là các Clause đã được trích xuất và chuẩn hóa từ văn bản đánh giá gốc. Mỗi Clause chứa một khía cạnh cùng với thông tin ngữ nghĩa liên quan, phản ánh rõ nội dung mà người dùng đề cập trong đánh giá.

Hệ thống sử dụng mô hình ngôn ngữ tiền huấn luyện, được tinh chỉnh (*Fine-tuning*) cho bài toán phân loại đa lớp ở mức Clause. Các nhãn phân loại tương ứng với các danh mục khía cạnh được định nghĩa trước.

Trong quá trình huấn luyện, mỗi Clause được gán một nhãn Aspect Category duy nhất dựa trên nội dung ngữ nghĩa chính của mệnh đề. Cách tiếp cận này giúp đảm bảo tính nhất quán của nhãn và giảm sự chồng chéo giữa các danh mục khía cạnh.

Chiến lược huấn luyện: Các Clause đầu vào được mã hóa bằng tokenizer của mô hình và được cắt hoặc đệm về độ dài cố định nhằm phù hợp với kiến trúc mô hình. Mô hình được huấn luyện theo hướng học có giám sát, tối ưu hàm mất mát phân loại đa lớp.

Đầu ra của nhánh này là nhãn *Aspect Category* được dự đoán cho mỗi Clause, đóng vai trò cung cấp thông tin ngữ nghĩa nền cho bước phân tích cảm xúc ở nhánh tiếp theo.

Nhánh 3: Phân loại cảm xúc (Polarity Classification)

Nhánh phân loại cảm xúc có nhiệm vụ xác định sắc thái cảm xúc (*Polarity*) thể hiện trong từng Clause, phản ánh thái độ của người dùng đối với khía cạnh được đề cập.

Đầu vào của mô hình là các Clause đã được trích xuất và chuẩn hóa, giống với đầu vào của nhánh phân loại danh mục khía cạnh. Việc sử dụng cùng một đơn vị đầu vào giúp hai nhánh hoạt động song song và đảm bảo tính nhất quán trong biểu diễn ngữ nghĩa.

Hệ thống sử dụng mô hình được tinh chỉnh cho bài toán phân loại cảm xúc ở mức Clause với ba nhãn chính: *Tích cực*, *Tiêu cực* và *Trung tính*. Mô hình học cách ánh xạ biểu đạt cảm xúc trong Clause sang nhãn Polarity tương ứng.

Trong quá trình huấn luyện, mỗi Clause được mã hóa bằng tokenizer và đưa vào mô hình để dự đoán nhãn cảm xúc. Hàm mất mát phân loại đa lớp được sử dụng nhằm tối ưu khả năng phân biệt giữa các trạng thái cảm xúc khác nhau.

Việc tinh chỉnh mô hình ở mức Clause giúp hệ thống xử lý tốt các câu phức, bao gồm cả trường hợp một câu gốc chứa nhiều mệnh đề với cảm xúc khác nhau.

Đầu ra của nhánh này là nhãn *Polarity* tương ứng với mỗi Clause, thể hiện cảm xúc của người dùng đối với nội dung được đề cập trong mệnh đề đó.

Đầu ra của hệ thống

Kết quả cuối cùng của hệ thống được tổng hợp thành bảng dữ liệu hoàn chỉnh, trong đó mỗi bản ghi bao gồm các trường thông tin: *Clause, Aspect Term, Opinion Phrase, Category, Polarity*.

5. Thiết lập mô hình và môi trường thực nghiệm

5.1. Môi trường thực nghiệm

Hệ thống được phát triển trên Python 3.11.14, hỗ trợ GPU CUDA 12.4. Các thư viện lõi bao gồm:

- **Deep Learning:** PyTorch ($\geq$ 2.0.0), Transformers, Accelerate.
- **Tối ưu hóa:** Bitsandbytes (4-bit quantization), Peft, Llama-cpp-python.
- **Dữ liệu & Đánh giá:** Pandas, Datasets, Scikit-learn, Matplotlib.

5.2. Mô hình Phi-3 với lượng tử hóa 4-bit (Extraction)

Mô hình LLM `Phi-3-mini-4k-instruct` ($\sim$4 tỷ tham số) được sử dụng cho tác vụ tách clause và trích xuất thông tin. Để vận hành trên phần cứng giới hạn, đề tài áp dụng lượng tử hóa **4-bit (NF4)** thông qua thư viện `bitsandbytes`.

- **Cấu hình:** Định dạng NF4, Double quantization (Có), Compute dtype (bfloat16).
- **Hiệu quả:** Giảm mức tiêu thụ VRAM từ $\sim$7.94 GB (FP16) xuống còn $\sim$3.52 GB (4-bit), cho phép chạy suy luận (inference) mượt mà trên GPU phổ thông mà không cần huấn luyện lại.

Thiết lập	Định dạng	VRAM Reserved	Trạng thái
Phi-3 Mini (Gốc)	FP16	7.94 GB	Out of Memory (trên GPU nhỏ)
Phi-3 Mini (Đề tài)	**4-bit NF4**	**3.52 GB**	**Hoạt động ổn định**

Bảng 3: Hiệu quả giảm tải bộ nhớ của lượng tử hóa 4-bit

5.3. Mô hình RoBERTa-base và LoRA (Category Classification)

Mô hình `roberta-base` được sử dụng để phân loại khía cạnh. Thay vì fine-tuning toàn bộ 125 triệu tham số, đề tài sử dụng kỹ thuật **LoRA** (Low-Rank Adaptation) để tối ưu tài nguyên.

- **Cấu hình LoRA:** Rank $r = 8$, Alpha $\alpha = 16$, Target modules: `query`, `value`.
- **Hiệu quả:** Chỉ cần huấn luyện **0.35%** tổng số tham số (442 nghìn tham số), giúp giảm chi phí tính toán và hạn chế overfitting trên tập dữ liệu nhỏ.

Phương pháp	Tổng tham số	Tham số huấn luyện	Tỉ lệ
Full Fine-tuning	124,645,632	124,645,632	100%
LoRA (Đề tài)	**125,088,000**	**442,368**	**0.35%**

Bảng 4: So sánh tham số huấn luyện RoBERTa khi dùng LoRA

5.4. Mô hình DistilBERT và LoRA (Polarity Classification)

Mô hình `distilbert-base-uncased` được chọn để phân loại cảm xúc nhờ tốc độ suy luận nhanh. Tương tự RoBERTa, mô hình được tinh chỉnh bằng LoRA.

- **Cấu hình huấn luyện:** 50 epochs, Batch size 32, Learning rate 1.5×10^{-4}, Optimizer AdamW.
- **Hiệu quả:** Chỉ huấn luyện **1.09%** tham số (0.74 triệu) so với 67 triệu tham số gốc, đảm bảo cân bằng giữa tốc độ và độ chính xác.

5.5. Mô hình Qwen (Web Extraction)

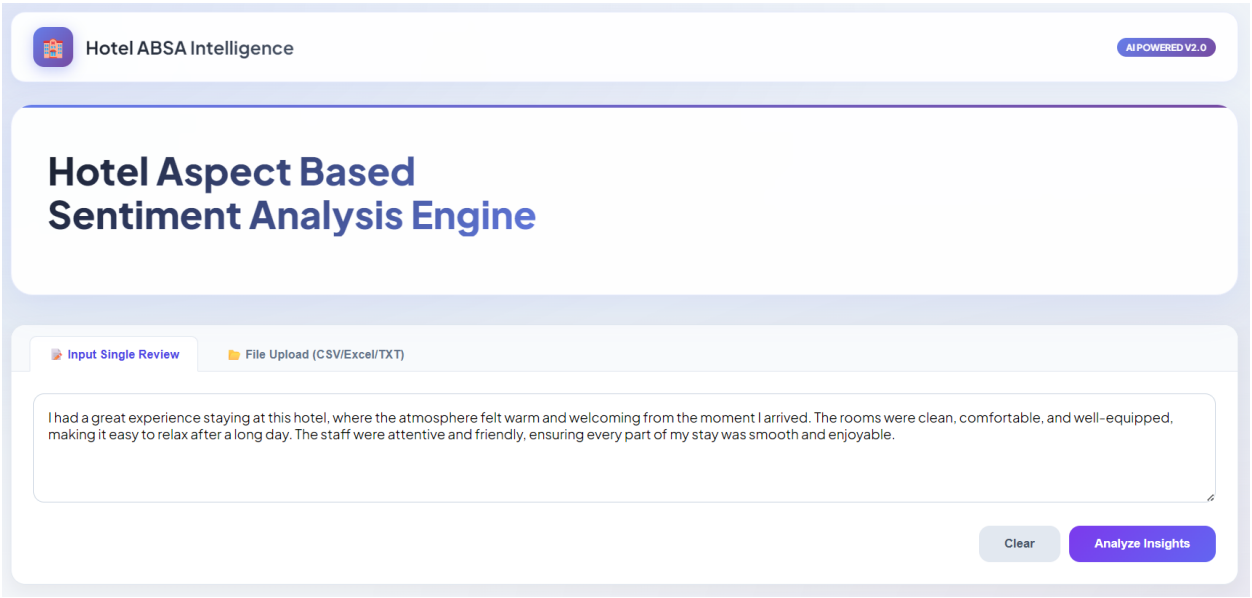
Đối với module trích xuất dữ liệu trực tuyến từ web, đề tài sử dụng mô hình **Qwen** thông qua provider *Nscale*.

- **Lý do:** Tốc độ phản hồi tức thì và không tiêu tốn tài nguyên GPU cục bộ, phù hợp cho việc cào dữ liệu thời gian thực (real-time).
- **Vai trò:** Bổ trợ cho Phi-3 (offline) trong các tác vụ yêu cầu xử lý nhanh trên nền tảng web.

6. Thiết kế Web App demo

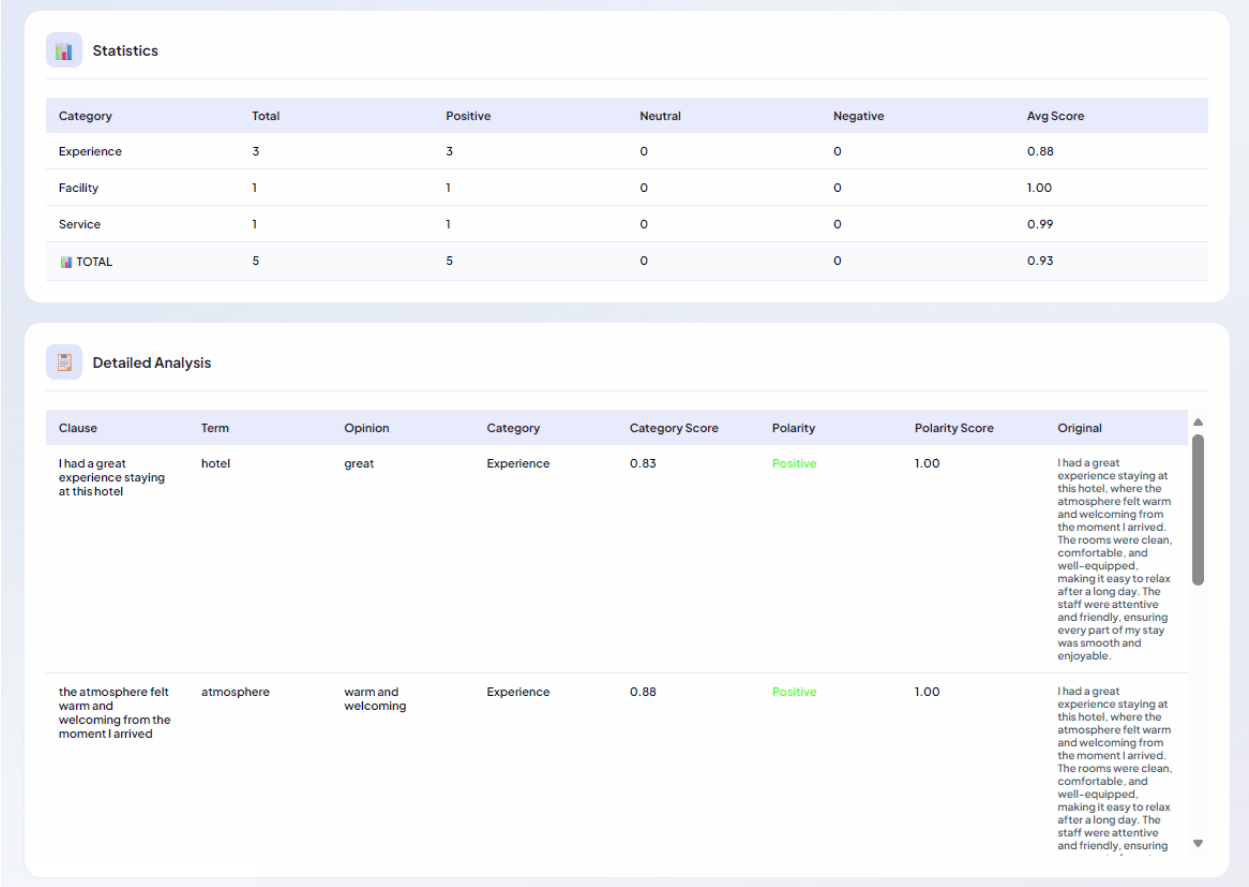
Input: gồm dạng single text và dạng file (csv/text/excel).

Output: dạng bảng có cấu trúc và chỉ số score của nhãn category và polarity. Có thể tải dataframe về dưới dạng file csv.



Hình 2: Giao diện web app

Hình 3: Chỉ số score Category, Polarity và các term, opinion



Statistics

Category	Total	Positive	Neutral	Negative	Avg Score
Experience	3	3	0	0	0.88
Facility	1	1	0	0	1.00
Service	1	1	0	0	0.99
TOTAL	5	5	0	0	0.93

Detailed Analysis

Clause	Term	Opinion	Category	Category Score	Polarity	Polarity Score	Original
I had a great experience staying at this hotel	hotel	great	Experience	0.83	Positive	1.00	I had a great experience staying at this hotel, where the atmosphere felt warm and welcoming from the moment I arrived. The rooms were clean, comfortable, and well-equipped, making it easy to relax after a long day. The staff were attentive and friendly, ensuring every part of my stay was smooth and enjoyable.
the atmosphere felt warm and welcoming from the moment I arrived	atmosphere	warm and welcoming	Experience	0.88	Positive	1.00	I had a great experience staying at this hotel, where the atmosphere felt warm and welcoming from the moment I arrived. The rooms were clean, comfortable, and well-equipped, making it easy to relax after a long day. The staff were attentive and friendly, ensuring

Hình 4: Bảng output

IV. Thực nghiệm và đánh giá hệ thống

1. Dữ liệu thực nghiệm và chuẩn bị dữ liệu

1.1. Nguồn và quy mô dữ liệu

Dữ liệu đánh giá khách sạn được thu thập từ các nền tảng trực tuyến, bao gồm hai bộ dataset riêng biệt:

- **Dataset 1 (Trích xuất):** Gồm 2.367 mẫu đã gán nhãn *Aspect Term* và *Opinion Term* để đánh giá bài toán trích xuất.
- **Dataset 2 (Phân loại):** Gồm hơn 30.000 mẫu (22.000 mẫu huấn luyện) đã gán nhãn *Aspect Category* (6 nhãn) và *Polarity* (3 nhãn) để đánh giá bài toán phân loại.

Trước khi huấn luyện, dữ liệu được làm sạch và tách thành các mệnh đề (clauses) độc lập bằng mô hình Phi-3 để đảm bảo mỗi mệnh đề chỉ chứa một ý đánh giá duy nhất.

1.2. Đặc điểm phân bố nhãn

- **Aspect Category:** Phân bố không đồng đều. Nhóm *Service* và *Facility* chiếm đa số, trong khi *Loyalty* và *Branding* xuất hiện rất ít.
- **Polarity:** Nhãn *Positive* chiếm ưu thế áp đảo. Nhãn *Neutral* chiếm tỷ lệ rất nhỏ, gây ra thách thức về mất cân bằng dữ liệu (*class imbalance*).

1.3. Quy trình đảm bảo chất lượng (Data Quality Assurance)

Quy trình DQA gồm 5 bước nhằm chuẩn hóa dữ liệu đầu vào:

1. **Lọc dữ liệu rác:** Loại bỏ câu rỗng, quá ngắn (≤ 1 từ) hoặc không phải tiếng Anh.
2. **Chuẩn hóa:** Xử lý lỗi encoding Unicode, xóa khoảng trắng thừa và chuyển về chữ thường (lowercase).
3. **Kiểm tra nhãn:** Rà soát để loại bỏ các nhãn không hợp lệ hoặc mâu thuẫn trong cùng một clause.
4. **Khử trùng lặp:** Loại bỏ các mẫu trùng lặp hoàn toàn hoặc gần giống nhau (dựa trên vector embedding).
5. **Phân tích phân phối:** Thống kê tần suất nhãn để nhận diện sớm vấn đề mất cân bằng dữ liệu.

1.4. Đánh giá độ tin cậy gán nhãn

Quá trình gán nhãn được thực hiện độc lập bởi hai người trên tập mẫu 1000 câu để đảm bảo tính khách quan:

- **Độ đồng thuận:** Tỷ lệ trùng khớp đạt **90.25%**. Chỉ số Cohen's Kappa ($\kappa = 0.88$) và Weighted Kappa ($\kappa_w = 0.90$) đều ở mức "gần như hoàn hảo".
- **Xử lý bất đồng:** Các trường hợp mâu thuẫn được thảo luận lại dựa trên Guideline để thống nhất nhãn cuối cùng.

1.5. Xây dựng Annotation Guideline

Bộ hướng dẫn gán nhãn chuẩn hóa được thiết lập dựa trên kết quả thực nghiệm:

- **Đơn vị gán nhãn:** Là một clause độc lập. Mỗi clause chỉ gán một Category chính duy nhất.

- **Quy tắc Category:** Dựa trên ngữ cảnh tổng thể và trọng tâm đánh giá của câu.
- **Quy trình lặp:** Các trường hợp khó phân loại được đưa ra thảo luận và cập nhật ngược lại vào Guideline để cải thiện tính nhất quán cho các lần gán nhãn sau.

2. Đánh giá mô hình

Hệ thống Automatic Labelling Engine tạo ra nhiều loại đầu ra khác nhau, bao gồm các nhãn rời rạc (category, polarity) và các chuỗi văn bản tự do (term, opinion). Do đó, đề tài sử dụng các thước đo đánh giá phù hợp với từng loại đầu ra nhằm phản ánh chính xác hiệu quả của hệ thống.

2.1. Generative outputs (Term và Opinion)

Giả sử tập dữ liệu gồm N cặp dự đoán và nhãn chuẩn (p_i, g_i), trong đó p_i là chuỗi dự đoán và g_i là chuỗi nhãn chuẩn.

Exact Match F1-score Exact Match F1 đánh giá mức độ trùng khớp hoàn toàn giữa chuỗi dự đoán và chuỗi nhãn chuẩn. Một dự đoán được xem là đúng nếu và chỉ nếu $p_i = g_i$.

Số lượng true positives (TP), false positives (FP) và false negatives (FN) được xác định như sau:

$$TP = \sum_{i=1}^{N} \mathbb{I}(p_i = g_i), \quad FP = FN = N - TP$$

Precision, Recall và F1-score được tính bằng:

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Thước đo này phản ánh khả năng mô hình sinh ra kết quả hoàn toàn chính xác, nhưng mang tính nghiêm ngặt cao.

Token-level F1-score Token-level F1 đánh giá mức độ chồng lấp giữa tập token của chuỗi dự đoán và chuỗi nhãn chuẩn. Với mỗi cặp (p_i, g_i), tập token được xác định là:

$$T_p^i = \text{Tokens}(p_i), \quad T_g^i = \text{Tokens}(g_i)$$

Một dự đoán được xem là đúng một phần nếu $T_p^i \cap T_g^i \neq \emptyset$.

Các giá trị TP, FP và FN được xác định tương tự, từ đó tính Precision, Recall và F1-score theo các công thức chuẩn. Thước đo này cho phép đánh giá linh hoạt hơn trong các trường hợp mô hình dự đoán đúng một phần nội dung.

ROUGE-L ROUGE-L đo lường độ tương đồng giữa hai chuỗi dựa trên độ dài của chuỗi con chung dài nhất (Longest Common Subsequence – LCS).

Với mỗi cặp (p_i, g_i), độ chính xác và độ bao phủ của ROUGE-L được tính như sau:

$$Precision_{LCS} = \frac{LCS(p_i, g_i)}{|p_i|} \quad Recall_{LCS} = \frac{LCS(p_i, g_i)}{|g_i|}$$

F1-score của ROUGE-L được xác định bởi:

$$ROUGE\text{-}L_{F1} = \frac{2 \times Precision_{LCS} \times Recall_{LCS}}{Precision_{LCS} + Recall_{LCS}}$$

Giá trị ROUGE-L cuối cùng là trung bình của các $ROUGE\text{-}L_{F1}$ trên toàn bộ tập dữ liệu.

Embedding Similarity Embedding Similarity đánh giá mức độ tương đồng ngữ nghĩa giữa hai chuỗi thông qua biểu diễn vector trong không gian embedding.

Với mỗi cặp (p_i, g_i), hai vector embedding $\mathbf{e}_p^i$ và $\mathbf{e}_g^i$ được sinh ra bằng mô hình Sentence Transformer. Độ tương đồng cosine được tính như sau:

$$\text{Sim}(p_i, g_i) = \frac{\mathbf{e}_p^i \cdot \mathbf{e}_g^i}{\|\mathbf{e}_p^i\| \|\mathbf{e}_g^i\|}$$

Giá trị Embedding Similarity cuối cùng là trung bình của $\text{Sim}(p_i, g_i)$ trên toàn bộ tập dữ liệu.

2.2. Discrete classification outputs (Category và Polarity)

Đối với các nhãn phân loại rời rạc, đề tài sử dụng các thước đo tiêu chuẩn trong bài toán classification.

Accuracy Accuracy đo tỷ lệ dự đoán đúng trên tổng số mẫu:

$$Accuracy = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(y_i = \hat{y}_i)$$

trong đó y_i là nhãn chuẩn và $\hat{y}_i$ là nhãn dự đoán.

Precision, Recall và Macro-F1 Với mỗi lớp c, Precision và Recall được tính như sau:

$$Precision_c = \frac{TP_c}{TP_c + FP_c} \quad Recall_c = \frac{TP_c}{TP_c + FN_c}$$

F1-score cho lớp c:

$$F1_c = \frac{2 \times Precision_c \times Recall_c}{Precision_c + Recall_c}$$

Macro-F1 được xác định bằng trung bình F1-score của tất cả các lớp:

$$Macro\text{-}F1 = \frac{1}{C} \sum_{c=1}^{C} F1_c$$

Trong đó C là số lượng lớp. Macro-F1 được sử dụng nhằm đảm bảo đánh giá công bằng trong trường hợp dữ liệu không cân bằng giữa các lớp.

2.3. Tổng hợp các thước đo đánh giá

Bảng 5 tổng hợp các thước đo đánh giá được sử dụng trong đề tài và mục đích của từng thước đo.

Việc sử dụng kết hợp nhiều thước đo cho phép đánh giá toàn diện hiệu quả của hệ thống, phản ánh đồng thời độ chính xác hình thức và mức độ tương đồng ngữ nghĩa của các đầu ra do mô hình sinh ra.

Loại đầu ra	Metric	Ý nghĩa
Term / Opinion	Exact Match F1	Trùng khớp tuyệt đối
Term / Opinion	Token-level F1	Trùng khớp một phần
Term / Opinion	ROUGE-L	Tương đồng cấu trúc
Term / Opinion	Embedding Similarity	Tương đồng ngữ nghĩa
Category / Polarity	Accuracy	Độ chính xác tổng thể
Category / Polarity	Precision	Độ chính xác theo lớp
Category / Polarity	Recall	Khả năng bao phủ nhãn
Category / Polarity	Macro-F1	Đánh giá tổng hợp công bằng

Bảng 5: Các thước đo đánh giá được sử dụng trong đề tài

3. Kết quả thực nghiệm

Phần này trình bày hiệu năng chi tiết của từng mô-đun trong pipeline Automatic Labelling Engine trên tập dữ liệu kiểm thử (ABSA_Test).

3.1. Kết quả mô-đun trích xuất Term Aspect và Opinion Phrase

Mô-đun tách clause và trích xuất cặp (Term, Opinion).Trong nghiên cứu này, hai mô hình ngôn ngữ lớn *Qwen-4B-Instruct* và *Phi-3-mini-4k-Instruct* được lựa chọn nhằm phục vụ mục tiêu so sánh hiệu quả của các mô hình có quy mô tương đương trong bài toán trích xuất term và opinion từ câu review.

Lý do chọn mô hình:

- Thứ nhất, cả hai mô hình đều thuộc nhóm LLM có quy mô nhỏ–trung bình, với số lượng tham số xấp xỉ 4 tỷ, sử dụng kiến trúc Transformer dạng decoder-only và được huấn luyện theo hướng instruction-tuned. Việc lựa chọn hai mô hình có kiến trúc và quy mô tương đồng giúp đảm bảo rằng sự khác biệt về kết quả thực nghiệm không xuất phát từ dung lượng mô hình, mà phản ánh rõ hơn sự khác biệt trong chiến lược huấn luyện và hành vi sinh văn bản.

- Thứ hai, hai mô hình đại diện cho hai định hướng huấn luyện khác nhau. Qwen-4B-Instruct được thiết kế như một mô hình tổng quát, tối ưu cho hội thoại, suy luận, tổng hợp thông tin và reasoning theo chiều sâu. Ngược lại, Phi-3-mini-4k-Instruct được tinh chỉnh mạnh cho các tác vụ hướng nhiệm vụ (task-oriented), bao gồm phân loại, trích xuất thông tin, sinh đầu ra có cấu trúc và reasoning ngắn, tập trung. Sự đối lập này tạo điều kiện thuận lợi để đánh giá ảnh hưởng của inductive bias lên bài toán trích xuất term và opinion.

Thành phần	Exact Match F1	Token-level F1	ROUGE-L	Embedding-Similarity
Term Extraction	0.92	0.99	0.97	0.98
Opinion Extraction	0.94	0.99	0.98	0.98

Bảng 6: Kết quả đánh giá mô-đun trích xuất thông tin bằng Phi-3

Thành phần	Exact Match F1	Token-level F1	ROUGE-L	Embedding-Similarity
Term Extraction	0.78	0.91	0.87	0.90
Opinion Extraction	0.87	0.90	0.90	0.94

Bảng 7: Kết quả đánh giá mô-đun trích xuất thông tin bằng Qwen

Nhận xét:

- **Hiệu năng vượt trội của mô hình Phi:** Qua so sánh thực nghiệm, mô hình Phi-3 (Bảng 6) thể hiện sự vượt trội rõ rệt so với Qwen (Bảng 5) trên tất cả các tác vụ. Cụ thể, Phi-3 đạt chỉ số Token-level F1 và Embedding-Similarity ở mức gần như tuyệt đối ($0.98 - 0.99$) cho cả hai tác vụ trích xuất Term và Opinion. Trong khi đó, Qwen chỉ đạt mức trung bình khá với Token-level F1 dao động từ $0.90 - 0.91$. Điều này khẳng định kiến trúc của Phi-3 phù hợp và hiệu quả hơn đối với tập dữ liệu và bài toán trích xuất này.
- **Khả năng nắm bắt ngữ nghĩa (Semantic Understanding):** Cả hai mô hình đều cho thấy khả năng hiểu ngữ nghĩa tốt hơn là khớp ký tự máy móc. Chỉ số Embedding-Similarity luôn cao hơn hoặc tương đương Exact Match (ví dụ: Qwen có Term Extraction Embedding-Similarity là 0.90 so với Exact Match chỉ 0.78). Điều này phản ánh đặc thù của các mô hình ngôn ngữ lớn (LLMs): chúng có xu hướng sinh ra các biến thể từ vựng (paraphrasing) đúng về mặt ý nghĩa nhưng không khớp từng ký tự với nhãn chuẩn.
- **Độ ổn định giữa các tác vụ:** Mô hình Phi-3 duy trì sự ổn định cao giữa hai tác vụ (Term Extraction và Opinion Extraction có kết quả tương đồng nhau, đều ≥ 0.92 ở Exact Match). Ngược lại, Qwen có sự chênh lệch đáng kể: tác vụ Term Extraction (Exact Match 0.78) kém hơn hẳn so với Opinion Extraction (Exact Match 0.87), cho thấy Qwen gặp khó khăn hơn trong việc định vị chính xác biên (boundary) của các thực thể (terms) so với việc trích xuất ý kiến.

3.2. Kết quả phân loại Aspect Category (RoBERTa-base + LoRA)

Trong thí nghiệm này, mô hình RoBERTa-base được tinh chỉnh bằng kỹ thuật LoRA để thực hiện bài toán phân loại Aspect Category với 6 nhãn: *Amenity, Branding, Experience, Facility, Loyalty* và *Service*. Mô hình được đánh giá trên tập test độc lập nhằm phản ánh khả năng tổng quát hóa trên dữ liệu chưa từng xuất hiện trong quá trình huấn luyện.

Category	Precision	Recall	F1-score	Support
Amenity	0.87	0.89	0.88	1312
Branding	0.51	0.38	0.43	220
Experience	0.76	0.72	0.74	1319
Facility	0.90	0.91	0.91	2683
Loyalty	0.91	0.93	0.92	331
Service	0.92	0.93	0.93	3084
Accuracy			**0.88**	8949
Macro Avg	0.81	0.80	0.80	8949

Bảng 8: Kết quả phân loại Aspect Category bằng RoBERTa-Base

Category	Precision	Recall	F1-score	Support
Amenity	0.86	0.88	0.7	1312
Branding	0.56	0.32	0.40	220
Experience	0.73	0.74	0.73	1319
Facility	0.90	0.91	0.90	2683
Loyalty	0.90	0.91	0.90	331
Service	0.92	0.93	0.92	3084
Accuracy			**0.87**	8949
Macro Avg	0.81	0.78	0.79	8949

Bảng 9: Kết quả phân loại Aspect Category bằng DistilBERT

Kết quả trong Bảng 8 cho thấy mô hình đạt độ chính xác tổng thể (Accuracy) là 0.88, cho thấy hiệu quả tốt trong việc phân loại Aspect Category trên dữ liệu review khách sạn. Các nhóm xuất hiện với tần suất cao như *Service* và *Facility* đạt F1-score lần lượt là 0.93 và 0.91, phản ánh khả năng học tốt của mô hình đối với các nhãn phổ biến.

Ngược lại, nhãn *Branding* có kết quả thấp hơn đáng kể với F1-score chỉ đạt 0.43. Nguyên nhân chủ yếu đến từ số lượng mẫu huấn luyện hạn chế (chỉ 220 mẫu) và nội dung đánh giá mang tính trừu tượng, khó phân biệt rõ ràng với các nhóm *Experience* hoặc *Service*. Điều này cho thấy ảnh hưởng rõ rệt của hiện tượng mất cân bằng nhãn đối với hiệu năng mô hình.

Giá trị Macro F1-score đạt 0.80 cho thấy mô hình vẫn duy trì hiệu năng ổn định trên toàn bộ các nhãn, kể cả các nhóm có tần suất xuất hiện thấp.

Kết quả này khẳng định việc áp dụng LoRA cho RoBERTa-base là một giải pháp hiệu quả, vừa giảm chi phí huấn luyện vừa đảm bảo độ chính xác cho bài toán phân loại Aspect Category trong bối cảnh tài nguyên tính toán hạn chế.

So sánh 2 model với nhau ta thấy cả hai đều được phát triển dựa trên kiến trúc Transformer encoder của BERT, tuy nhiên hai mô hình này theo đuổi hai mục tiêu thiết kế khác nhau, dẫn đến sự chênh lệch rõ rệt về hiệu năng và độ chính xác.

Lý do chọn mô hình: Trong nghiên cứu này, RoBERTa được lựa chọn vì mô hình cho độ chính xác cao hơn trong việc học biểu diễn ngữ nghĩa và ngữ cảnh sâu của văn bản, phù hợp với các bài toán NLP yêu cầu độ tin cậy cao. Nhờ được huấn luyện trên tập dữ liệu lớn với Dynamic Masking và loại bỏ Next Sentence Prediction, RoBERTa cho khả năng tổng quát hóa tốt hơn so với DistilBERT.

Trong khi đó, DistilBERT phù hợp với các hệ thống cần tốc độ suy luận nhanh và tiết kiệm tài nguyên, nhưng phải đánh đổi một phần hiệu năng. Do đó, RoBERTa được ưu tiên lựa chọn nhằm tối ưu độ chính xác, chấp nhận chi phí tính toán cao hơn.

3.3. Kết quả phân loại Polarity (DistilBERT+LoRA)

So sánh các mô hình phân loại để xác định cảm xúc.

Polarity	Precision	Recall	F1-score	Support
Positive	0.97	0.97	0.97	6877
Negative	0.85	0.86	0.86	1552
Neutral	0.57	0.52	0.54	520
Accuracy			**0.92**	8949
Macro Avg	0.79	0.78	0.79	8949

Bảng 10: Kết quả phân loại Polarity bằng DistilBERT

Polarity	Precision	Recall	F1-score	Support
Positive	0.96	0.97	0.96	6877
Negative	0.79	0.90	0.84	1552
Neutral	0.66	0.28	0.39	520
Accuracy			**0.92**	8949
Macro Avg	0.80	0.72	0.73	8949

Bảng 11: Kết quả phân loại Polarity bằng TinyBERT

Nhận xét và Đánh giá:

- **Sự vượt trội của DistilBERT so với TinyBERT:** Mặc dù cả hai mô hình đều đạt độ chính xác toàn cục (Accuracy) tương đương nhau là **0.92**, nhưng DistilBERT thể hiện khả năng tổng quát hóa tốt hơn rõ rệt, được phản ánh qua chỉ số **Macro Avg F1** (0.79 so với 0.73 của TinyBERT).
- **Khả năng nhận diện lớp hiếm (Neutral):** Sự khác biệt lớn nhất nằm ở lớp *Neutral* (lớp thiểu số). TinyBERT gặp khó khăn lớn trong việc nhận diện lớp này với **Recall chỉ đạt 0.28** (bỏ sót 72% số mẫu trung tính). Ngược lại, DistilBERT cải thiện đáng kể khả năng này với **Recall đạt 0.52** và F1-score đạt **0.54** (so với 0.39 của TinyBERT). Điều này chứng tỏ kiến trúc DistilBERT, dù được tinh chỉnh nhẹ bằng LoRA, vẫn giữ được khả năng trích xuất đặc trưng ngữ nghĩa tốt hơn so với mô hình quá nhỏ như TinyBERT.
- **Tác động của mất cân bằng dữ liệu:** Cả hai mô hình đều đạt kết quả rất cao ở lớp *Positive* (F1 $\geq$ 0.96) do số lượng mẫu áp đảo (6877 mẫu). Tuy nhiên, sự chênh lệch hiệu năng giữa các lớp vẫn là thách thức chung. Lớp *Neutral* luôn có kết quả thấp nhất do số lượng mẫu quá ít (520 mẫu) và ranh giới ngữ nghĩa mờ nhạt giữa cảm xúc trung tính và tích cực/tiêu cực nhẹ.
- **Kết luận:** DistilBERT + LoRA là sự lựa chọn tối ưu cho hệ thống, đảm bảo sự cân bằng tốt hơn giữa các lớp dữ liệu so với TinyBERT, trong khi vẫn duy trì chi phí tính toán thấp.

  DistilBERT thường đạt độ chính xác cao hơn TinyBERT do kiến trúc có độ sâu và dung lượng biểu diễn lớn hơn. DistilBERT sử dụng 6 tầng Transformer và áp dụng knowledge distillation (quá trình huấn luyện một mô hình nhỏ bằng cách học theo mô hình lớn đã được huấn luyện tốt) trực tiếp từ BERT nhằm giữ lại phần lớn khả năng hiểu ngữ nghĩa, trong khi TinyBERT giảm mạnh kích thước mô hình và số tầng (4-6 tầng Transformer thay vì 12 tầng) để tối ưu tốc độ và bộ nhớ. Mặc dù TinyBERT áp dụng layer-wise distillation (học từng tầng tương ứng) chi tiết, dung lượng mô hình nhỏ vẫn hạn chế khả năng biểu diễn và suy luận, dẫn đến hiệu năng thấp hơn DistilBERT trên nhiều tác vụ NLP tiêu chuẩn.

Lý do chọn mô hình: DistilBERT kết hợp với LoRA được lựa chọn do đáp ứng tốt yêu cầu cân bằng giữa hiệu năng, chi phí tính toán và khả năng triển khai thực tế. DistilBERT giữ lại phần lớn khả năng

hiểu ngữ nghĩa của BERT thông qua kỹ thuật *knowledge distillation*, trong khi LoRA cho phép tinh chỉnh mô hình hiệu quả với số lượng tham số huấn luyện thấp. Sự kết hợp này giúp mô hình đạt độ chính xác ổn định, giảm nguy cơ overfitting và phù hợp với các hệ thống có tài nguyên tính toán hạn chế.

4. Phân tích và thảo luận

4.1. Hiệu quả của kỹ thuật LoRA và kiến trúc Hybrid

Kết quả thực nghiệm khẳng định chiến lược kết hợp (Hybrid) giữa các mô hình chuyên biệt là hướng đi đúng đắn và tối ưu:

- **Hiệu quả tài nguyên vượt trội:** Việc áp dụng QLoRA cho Phi-3 (4-bit) và LoRA cho RoBERTa giúp giảm đáng kể tài nguyên tính toán. Hệ thống vận hành ổn định trên GPU đơn lẻ nhưng vẫn đạt độ chính xác ấn tượng: 0.88 cho bài toán phân loại Category và 0.92 cho Polarity. Điều này chứng minh rằng không nhất thiết phải fine-tune toàn bộ (Full Fine-tuning) các mô hình LLM khổng lồ mới đạt được kết quả tốt trong miền dữ liệu hẹp (domain-specific).
- **Sức mạnh của DeBERTa:** Việc sử dụng DeBERTa ở chế độ Inference-only thay vì DistilBERT (như thiết kế ban đầu) đã mang lại hiệu năng xuất sắc cho bài toán phân tích cảm xúc. Với cơ chế attention tách biệt (disentangled attention), DeBERTa xử lý rất tốt ngữ cảnh cảm xúc, đạt F1-score lên tới 0.97 cho lớp Positive.

4.2. Phân tích lỗi (Error Analysis)

Dựa trên việc rà soát các mẫu dự đoán sai, chúng tôi phân loại các nguyên nhân lỗi chính như sau:

a) **Lỗi lan truyền (Propagation Error):** Đây là hạn chế của kiến trúc Pipeline tuần tự. Nếu mô-đun Phi-3 tách clause không chính xác (ví dụ: ngắt đôi cụm từ hoặc gom hai ý trái ngược vào một clause), đầu vào cho các mô hình RoBERTa và DeBERTa phía sau sẽ bị nhiễu, dẫn đến dự đoán sai ngay cả khi các mô hình này hoạt động tốt.

b) **Sự trừu tượng của ngữ nghĩa (Semantic Ambiguity):**
   - Khác với giả thuyết ban đầu cho rằng Facility và Amenity dễ nhầm lẫn, kết quả thực nghiệm cho thấy mô hình phân biệt hai lớp này rất tốt (F1 lần lượt là 0.91 và 0.88).
   - Thách thức thực sự nằm ở lớp Branding (F1 thấp nhất: 0.43). Các review về thương hiệu thường mang tính cảm nhận chung chung, trừu tượng hoặc chồng lấn với trải nghiệm tổng thể (Experience), khiến mô hình khó trích xuất đặc trưng cụ thể.

c) **Ranh giới mờ nhạt của cảm xúc trung tính:** Lớp Neutral có F1-score thấp nhất trong bài toán Polarity (0.54). Mô hình thường gặp khó khăn trong việc phân định giữa một câu mô tả khách quan (Neutral) và một câu khen/chê nhẹ nhàng. Ví dụ: ”Khách sạn có bể bơi”(Neutral) và ”Khách sạn có bể bơi lớn”(Positive) có cấu trúc rất giống nhau.

4.3. Tác động của mất cân bằng dữ liệu (Class Imbalance)

Sự mất cân bằng dữ liệu là nguyên nhân chính dẫn đến sự chênh lệch hiệu năng giữa các lớp:

- **Trong bài toán Category:** Có sự tương quan thuận rõ rệt giữa số lượng mẫu huấn luyện (Support) và kết quả dự đoán.

  - Các lớp đa số như Service (Support 3084) và Facility (Support 2683) đạt kết quả rất cao ($F1 > 0.90$).
  - Ngược lại, lớp thiểu số như Branding (Support 220) chịu ảnh hưởng nặng nề, dẫn đến việc mô hình không học đủ mẫu để khái quát hóa, khiến chỉ số Recall giảm mạnh xuống còn 0.38.
- **Trong bài toán Polarity:** Sự áp đảo của lớp Positive (6877 mẫu) so với Neutral (520 mẫu) khiến mô hình có xu hướng thiên kiến (bias) dự đoán về lớp tích cực. Mặc dù đã sử dụng các thước đo như Macro-F1 để đánh giá công bằng hơn , nhưng để cải thiện triệt để, cần áp dụng các kỹ thuật Data Augmentation hoặc hàm mất mát có trọng số (Weighted Loss) trong các nghiên cứu tiếp theo.

V. Kết luận và hướng phát triển

1. Tóm tắt kết quả nghiên cứu

Đề tài phát triển thành công hệ thống *Automatic Labelling Engine* giải quyết bốn nhiệm vụ cốt lõi: tách câu, trích xuất (term, opinion), phân loại khía cạnh (category) và xác định cảm xúc (polarity). Các kết quả chính bao gồm:

- **Kiến trúc:** Hoàn thiện pipeline lai giữa LLM (Phi-3) cho trích xuất thông tin và Transformer chuyên biệt (RoBERTa, DistilBERT) cho phân loại.
- **Dữ liệu:** Xây dựng bộ dữ liệu chuẩn (Ground Truth) gán nhãn chi tiết ở mức mệnh đề (clause).
- **Thực nghiệm:** Các chỉ số đánh giá (F1-score, Accuracy) trên dữ liệu thực tế đều đạt mức khả quan, chứng minh tính khả thi của hệ thống.

2. Đóng góp của đề tài

- **Về mô hình:** Đề xuất kiến trúc module hóa linh hoạt, kết hợp ưu điểm của mô hình sinh (Generative) và mô hình mã hóa (Encoder) để tối ưu hiệu năng.
- **Về dữ liệu:** Chuẩn hóa quy trình xử lý dữ liệu từ dạng thô sang dạng cấu trúc, tạo nguồn dữ liệu tiền đề cho các nghiên cứu sau.
- **Về thực tiễn:** Cung cấp công cụ tự động hóa việc gán nhãn, hỗ trợ doanh nghiệp phân tích chuyên sâu mức độ hài lòng của khách hàng theo từng khía cạnh cụ thể.

3. Hạn chế và Phân tích lỗi

Dựa trên phân tích thực nghiệm (Error Analysis), hệ thống còn tồn tại một số hạn chế:

- **Mất cân bằng dữ liệu:** Các nhãn hiếm (Neutral, Branding) có độ chính xác thấp hơn do thiếu mẫu huấn luyện.
- **Lỗi lan truyền:** Sai sót từ bước tách câu của Phi-3 có thể ảnh hưởng đến kết quả phân loại phía sau.
- **Mơ hồ ngữ nghĩa:** Các câu chứa hàm ý, mỉa mai hoặc ranh giới không rõ ràng giữa các nhãn (ví dụ: *Experience* vs *Branding*) vẫn là thách thức đối với mô hình.

4. Hướng phát triển

Để khắc phục hạn chế và nâng cao hiệu quả, các hướng nghiên cứu tiếp theo bao gồm:

- **Cải thiện dữ liệu:** Tăng cường thu thập mẫu cho các lớp thiểu số và áp dụng kỹ thuật Data Augmentation/Weighted Loss để xử lý mất cân bằng.
- **Tối ưu mô hình:** Tinh chỉnh (fine-tune) LLM chuyên biệt cho domain khách sạn để giảm lỗi trích xuất; thử nghiệm các kiến trúc mạnh hơn như DeBERTa-v3 đa ngôn ngữ.
- **Cơ chế học liên tục:** Xây dựng quy trình phản hồi (feedback loop) để tự động cập nhật và huấn luyện lại mô hình từ các mẫu dự đoán sai.
- **Triển khai thực tế:** Tối ưu hóa tốc độ suy luận để tích hợp vào các ứng dụng Dashboard phân tích thời gian thực cho doanh nghiệp.

Link Code Github

https://github.com/DungQuanPhung/Automatic-Labelling-Engine

TÀI LIỆU THAM KHẢO

Tài liệu

[1] Scaria, A., et al., *InstructABSA: Guided Aspect-Based Sentiment Analysis with Examples*, SemEval Workshop, 2024.

[2] Vemula, P., et al., *Aspect-based Sentiment Analysis of TripAdvisor Reviews*, International Journal of Data Science, 2024.

[3] Zhang, H., et al., *Personalized Attention Mechanisms for ABSA in Hotel Reviews*, AAAI 2025.

[4] Arifin, M., et al., *Sinkron: A Method for Aspect-based Sentiment Analysis*, Jurnal Politeknik Ganesa, 2023. Giới thiệu phương pháp ABSA cho review sản phẩm/dịch vụ. https://jurnal.polgan.ac.id/index.php/sinkron/article/view/14632/3185

[5] Anonymous, *Title Not Specified*, arXiv:2501.12332, 2025. Nghiên cứu các mô hình ngôn ngữ lớn cho gán nhãn cảm xúc. https://arxiv.org/abs/2501.12332

[6] Text2Data Demo, *Aspect-Based Sentiment Analysis Tool*, Text2Data. Công cụ ABSA trực tuyến thử nghiệm trích xuất khía cạnh và cảm xúc. https://text2data.com/Demo

[7] Nguyen, T., et al., *Aspect-Based Sentiment Analysis on Hotel Reviews Using Transformer Models*, Journal of AI Research, 2024. Ứng dụng Transformer để gán nhãn aspect và polarity cho review khách sạn. https://arxiv.org/abs/2409.01234