TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

**KHOA ĐIỆN TỬ**

**Bộ môn: Công nghệ Thông tin**.

**BÀI TẬP KẾT THÚC MÔN HỌC**

MÔN HỌC

**KHOA HỌC DỮ LIỆU**

  Sinh viên: Đ INH NGUYỄN HOÀNG VŨ

  Lớp: K57KMT

Giáo viên giảng dạy: NGUYỄN VĂN HUY

Link GitHub: https://github.com/dnhnghv/TL\_KhoaHocDuLieu

**THÁI NGUYÊN 2025**

|  |  |
| --- | --- |
| **TRƯỜNG ĐHKTCN** | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM** |
| **KHOA ĐIỆN TỬ** | ***Độc lập - Tự do - Hạnh phúc*** |

**BÀI TẬP KẾT THÚC MÔN HỌC**

**MÔN HỌC: KHOA HỌC DỮ LIỆU**

BỘ MÔN : CÔNG NGHỆ THÔNG TIN

*Sinh viên: Đinh Nguyễn Hoàng Vũ*………………………………………...

*Lớp*: *K57KMT*……………………. *Ngành: Kỹ thuật máy tính*……………

*Giáo viên hướng dẫn: Nguyễn Văn Huy*

*Ngày giao đề:20-5-2025*  *Ngày hoàn thành:30-5-2025*

*Tên đề tài:* Phân tích cảm xúc comment Facebook

*Yêu cầu:* Phân tích comment Facebook để xác định cảm xúc tích cực, tiêu cực.

*Đầu ra:* Thống kê tỷ lệ cảm xúc, visualization bằng Pie Chart.

**Các tính năng:**

* Phân tích ngôn ngữ tự nhiên (CountVectorizer, TF-IDF)
* Classification (Naive Bayes, Logistic Regression)
* Trực quan hóa dữ liệu cảm xúc

**Giáo Viên Hướng Dẫn**

*Ký và ghi rõ họ tên*

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

Thái Nguyên, ngày….tháng…..năm 20....

**GIÁO VIÊN HƯỚNG DẪN**

*(Ký ghi rõ họ tên)*

Mục Lục

[CHƯƠNG I: GIỚI THIỆU ĐẦU BÀI 8](#_Toc199491931)

[1.1. Đặt vấn đề 8](#_Toc199491932)

[1.2. Mục tiêu của dự án 8](#_Toc199491933)

[1.3. Các tính năng chính của hệ thống 9](#_Toc199491934)

[1.4. Thách thức 9](#_Toc199491935)

[1.5. Kiến thức đã vận dụng 10](#_Toc199491936)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 13](#_Toc199491937)

[2.1. Ngôn ngữ lập trình Python và cấu trúc dữ liệu cơ bản 13](#_Toc199491938)

[2.2. Xử lý ngôn ngữ tự nhiên (NLP) 13](#_Toc199491939)

[2.3. Xử lý dữ liệu với Pandas và NumPy 14](#_Toc199491940)

[2.4. Học máy (Machine Learning) 15](#_Toc199491941)

[2.5. Trực quan hóa dữ liệu với Matplotlib 16](#_Toc199491942)

[2.6. Facebook Graph API 16](#_Toc199491943)

[CHƯƠNG 3: THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH 18](#_Toc199491944)

[3.2. Chương trình chính 20](#_Toc199491945)

[CHƯƠNG 4. THỰC NGHIỆM VÀ KẾT LUẬN 27](#_Toc199491946)

[4.1. Thực nghiệm 27](#_Toc199491947)

[4.2. Kết luận 34](#_Toc199491948)

**Danh mục biểu đồ và đồ thị**

*Hình 3.1 biểu đồ phân cấp chức năng* 18

*Hình 4.1 Visual Studio Code* 24

*Hình 4.2 Python là gì?* 26

*Hình 4.3 Trang facebook mà sẽ lấy dữ liệu* 28

*Hình 4.4 bài viết dùng để lấy dữ liệu* 29

*Hình 4.5 Biểu đồ cảm xúc sau khi lấy bình luận từ Facebook* 30

*Hình 4.6 Bảng hiển thị chi tiết từng bình luận và nhãn cảm xúc* 32

**LỜI NÓI ĐẦU**

Em xin trân trọng giới thiệu báo cáo kết quả thực hiện đề tài “Phân tích cảm xúc từ bình luận Facebook” trong khuôn khổ môn Khoa học Dữ liệu. Đề tài nhằm xây dựng một hệ thống tự động thu thập bình luận từ Facebook, tiền xử lý văn bản, ứng dụng mô hình học máy để phân loại cảm xúc (tích cực, trung tính, tiêu cực) và hiển thị kết quả dưới dạng trực quan. Qua quá trình thực hiện, em đã được vận dụng kiến thức về xử lý ngôn ngữ tự nhiên, học máy, lập trình Python và tương tác API, đồng thời rèn luyện kỹ năng thiết kế, triển khai và đánh giá một hệ thống phần mềm hoàn chỉnh.

Em hy vọng báo cáo sẽ trình bày rõ ý tưởng, phương pháp và kết quả đạt được, cũng như những thách thức gặp phải trong quá trình làm đề tài. Mọi ý kiến góp ý của Thầy sẽ là động lực và hướng phát triển quan trọng cho em trong những nghiên cứu tiếp theo.

**LỜI CẢM ƠN**

Em xin chân thành cảm ơn Thầy đã tận tình hướng dẫn, chia sẻ kiến thức, kinh nghiệm quý báu trong suốt thời gian thực hiện đề tài. Sự hỗ trợ kịp thời của Thầy về phương pháp, công cụ, cũng như những góp ý mang tính xây dựng đã giúp em hoàn thiện báo cáo và sản phẩm. Em cũng xin cảm ơn các bạn trong nhóm đã cùng trao đổi, thảo luận và đồng hành với em trong suốt quá trình làm việc.

Em chân thành cảm ơn Thầy!

# CHƯƠNG I: GIỚI THIỆU ĐẦU BÀI

## **1.1. Đặt vấn đề**

Mạng xã hội, đặc biệt là Facebook, hiện nay là một trong những nền tảng phổ biến nhất để người dùng chia sẻ cảm xúc, quan điểm cá nhân và phản hồi đối với các sự kiện, sản phẩm, dịch vụ. Trong số đó, các bình luận (comments) trên các bài viết đóng vai trò quan trọng trong việc thể hiện cảm xúc thực tế của người dùng. Khai thác và phân tích những dữ liệu này mang lại giá trị thiết thực trong nhiều lĩnh vực như marketing, chăm sóc khách hàng, truyền thông và nghiên cứu xã hội học.

Tuy nhiên, việc xử lý và phân tích bình luận người dùng gặp nhiều thách thức do tính phi cấu trúc của dữ liệu văn bản, đặc biệt là tiếng Việt, với đặc điểm ngôn ngữ đa dạng, nhiều viết tắt, emoji và ngữ cảnh phức tạp. Do đó, bài toán phân loại cảm xúc trong bình luận Facebook là một bài toán thực tế và có ý nghĩa trong lĩnh vực Khoa học dữ liệu.

**Ý tưởng thực hiện:** xây dựng một phần mềm có thể lấy được comment trên facebook một cách trực tiếp, phân tích nội dung các comment dựa trên mô hình sentiment classification được huấn luyện trước nhằm phân tích dữ liệu đã thu thập được xem đâu là comment tích cực, tiêu cực, trung tính. Vẽ biểu đồ trực quan hóa dữ liệu đã thu thập và phân tích được nhằm đánh giá tổng quát và dễ dàng chất lượng cũng như tính chất của bài viết

## **1.2. Mục tiêu của dự án**

Dự án hướng tới xây dựng một hệ thống có khả năng:

* Thu thập dữ liệu bình luận từ một fanpage Facebook bất kỳ bằng cách sử dụng Facebook Graph API.
* Áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên và học máy để phân tích cảm xúc của từng bình luận, phân loại thành ba nhóm: tích cực, tiêu cực và trung tính.
* Trực quan hóa kết quả dưới dạng biểu đồ tròn (Pie Chart), thể hiện tỷ lệ cảm xúc tổng quan từ các bình luận.
* Cập nhật biểu đồ cảm xúc một cách tự động và liên tục mỗi 5 giây, phản ánh dữ liệu theo thời gian thực.

## **1.3. Các tính năng chính của hệ thống**

Hệ thống được thiết kế với các tính năng chính như sau:

* Tự động thu thập bình luận Facebook thông qua Facebook Graph API, có thể cấu hình fanpage cần theo dõi.
* Phân tích cảm xúc tích cực tiêu cực bằng mô hình học máy đã được huấn luyện trên tập dữ liệu có gán nhãn.
* Xây dựng giao diện web trực quan, hiển thị biểu đồ cảm xúc theo thời gian thực, đánh giá chất lượng bài viết thông qua phân tích cảm xúc của comment.
* Cập nhật biểu đồ cảm xúc tự động mỗi 5 giây, mô phỏng hệ thống phân tích online liên tục.

## **1.4. Thách thức**

Trong quá trình triển khai, em gặp phải một số thách thức tiêu biểu:

* Hạn chế về quyền truy cập dữ liệu từ Facebook: Việc cấu hình quyền hạn (scope), tạo Access Token đúng cách là bắt buộc nhưng tương đối phức tạp và dễ bị giới hạn bởi chính sách bảo mật.
* Xử lý dữ liệu văn bản tiếng Việt: Dữ liệu ngôn ngữ tự nhiên, đặc biệt là bình luận mạng xã hội, thường ngắn, phi cấu trúc, chứa nhiều từ viết tắt, ký hiệu và biểu tượng cảm xúc gây khó khăn cho việc phân tích cảm xúc.
* Huấn luyện và lựa chọn mô hình: Việc lựa chọn mô hình học máy phù hợp với dữ liệu tiếng Việt và đảm bảo tốc độ phân tích nhanh là một thách thức đáng kể.
* Xử lý dữ liệu thời gian thực: Đảm bảo hệ thống có thể cập nhật biểu đồ cảm xúc một cách liên tục, đồng bộ giữa dữ liệu đầu vào và hiển thị đầu ra đòi hỏi tối ưu về thiết kế kỹ thuật.

## **1.5. Kiến thức đã vận dụng**

Dự án phân tích cảm xúc từ bình luận Facebook là một ví dụ điển hình về việc ứng dụng toàn diện kiến thức trong môn học Khoa học dữ liệu, bao gồm:

*1. Introduction to NumPy:*

Thư viện NumPy được sử dụng để xử lý dữ liệu số, thực hiện các phép biến đổi vector, ma trận cần thiết trong bước chuẩn hóa dữ liệu đầu vào cho mô hình học máy. Nhóm vận dụng NumPy để xử lý nhanh và hiệu quả các mảng dữ liệu lớn.

*2. Data Manipulation with Pandas:*

Pandas là công cụ quan trọng trong việc tổ chức, lọc, và xử lý dữ liệu văn bản sau khi thu thập. Các thao tác như đọc dữ liệu, làm sạch, gán nhãn cảm xúc, xử lý cột đặc trưng đều được thực hiện với Pandas, giúp đơn giản hóa quy trình tiền xử lý.

*3. Text Preprocessing:*

Dữ liệu bình luận được xử lý bằng các kỹ thuật xử lý ngôn ngữ tự nhiên như chuyển văn bản thành chữ thường, loại bỏ dấu câu, stopwords, token hóa câu, chuẩn hóa từ. Đây là bước bắt buộc để mô hình học máy có thể hiểu được ngữ cảnh và cảm xúc trong văn bản.

*4. Feature Extraction:*

Sử dụng các phương pháp biến văn bản thành dạng vector như Bag-of-Words, TF-IDF hoặc Embedding, nhóm đã chuyển đổi dữ liệu văn bản thành các đặc trưng số để mô hình học máy có thể xử lý.

*5. Visualization with Matplotlib:*

Matplotlib được sử dụng để trực quan hóa dữ liệu phân tích cảm xúc thông qua biểu đồ tròn (Pie Chart), giúp người dùng dễ dàng theo dõi tỷ lệ bình luận tích cực, tiêu cực và trung tính. Việc trực quan hóa không chỉ phục vụ mục đích trình bày mà còn giúp đánh giá hiệu quả mô hình một cách trực quan.

*6. Machine Learning:*

Dự án áp dụng kiến thức về học máy để xây dựng mô hình phân loại cảm xúc. Các thuật toán như Logistic Regression, Naive Bayes hoặc mô hình hiện đại như BERT tiếng Việt được đưa vào huấn luyện. Nhóm đã thực hiện đánh giá mô hình với các chỉ số như Accuracy, Precision, Recall, F1-score.

*7. API Integration & Automation:*

Việc sử dụng Facebook Graph API yêu cầu hiểu cách gọi API, xử lý dữ liệu JSON và tự động cập nhật comment mới. Nhóm đã xây dựng đoạn mã có thể tự động hóa quy trình này, đóng vai trò như một pipeline khoa học dữ liệu hoàn chỉnh.

*8. Real-time Data Update:*

Triển khai cơ chế cập nhật biểu đồ cảm xúc mỗi 5 giây, phản ánh khả năng xử lý dữ liệu theo thời gian thực, đây là một yêu cầu thực tế thường gặp trong các hệ thống phân tích mạng xã hội.

*9. Data Pipeline Development:*

Xây dựng một pipeline toàn diện: Thu thập dữ liệu (Facebook API) → Tiền xử lý (Pandas, NLP) → Trích xuất đặc trưng (TF-IDF/BERT) → Huấn luyện mô hình → Trực quan hóa (Matplotlib + Web), phản ánh đúng quy trình khoa học dữ liệu.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

Trong chương trình phân tích cảm xúc từ bình luận Facebook, nhóm đã vận dụng nhiều kiến thức từ lĩnh vực khoa học dữ liệu, học máy và lập trình Python. Phần này trình bày các kiến thức chuyên môn cốt lõi được sử dụng để xây dựng hệ thống.

## **2.1. Ngôn ngữ lập trình Python và cấu trúc dữ liệu cơ bản**

Python được chọn làm ngôn ngữ phát triển chính vì tính linh hoạt, cộng đồng hỗ trợ rộng rãi và có nhiều thư viện chuyên biệt phục vụ cho học máy và phân tích dữ liệu.

Các cấu trúc dữ liệu cơ bản của Python như:

List: Dùng để lưu danh sách các bình luận, danh sách kết quả dự đoán,...

Dictionary: Dùng để ánh xạ giữa các nhãn cảm xúc và các chỉ số thống kê.

Tuple & Set: Hữu ích trong việc xử lý và loại bỏ trùng lặp khi lọc từ vựng, từ khóa.

Ngoài ra, nhóm còn vận dụng kỹ năng xử lý chuỗi, định nghĩa hàm, viết hàm xử lý dữ liệu, đọc/ghi file, điều kiện và vòng lặp,... để xây dựng chương trình một cách logic và hiệu quả.

## **2.2. Xử lý ngôn ngữ tự nhiên (NLP)**

Vì dữ liệu đầu vào của chương trình là bình luận tiếng Việt nên cần áp dụng các kỹ thuật tiền xử lý văn bản thuộc lĩnh vực Xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP). Một số bước quan trọng bao gồm:

*Làm sạch dữ liệu (Data Cleaning):*

Loại bỏ các ký tự đặc biệt như emoji, dấu câu, đường link, số.

Chuyển toàn bộ văn bản về chữ thường để thống nhất định dạng.

*Xóa bỏ khoảng trắng thừa.*

*Tách từ (Tokenization):*

Tiếng Việt không phân cách từ bằng dấu cách như tiếng Anh, vì vậy tokenization đóng vai trò cực kỳ quan trọng. Nhóm sử dụng thư viện `underthesea` hoặc `pyvi` để thực hiện tách từ chính xác hơn.

*Loại bỏ stopwords:*

Stopwords là những từ không mang nhiều ý nghĩa như: “là”, “và”, “thì”, “nhưng”... Việc loại bỏ giúp tăng độ tập trung vào các từ mang tính cảm xúc thực sự.

*Chuyển văn bản thành vector (Text Vectorization):*

Văn bản sau khi xử lý được chuyển thành dạng số bằng kỹ thuật TF-IDF (Term Frequency - Inverse Document Frequency).

Kỹ thuật này đánh giá tầm quan trọng của từ trong từng bình luận và trên toàn tập dữ liệu.

## **2.3. Xử lý dữ liệu với Pandas và NumPy**

**Pandas:** là thư viện mạnh mẽ dùng để thao tác với dữ liệu dạng bảng (DataFrame).

Nhóm sử dụng Pandas để đọc file dữ liệu, lọc những bình luận hợp lệ, phân nhóm bình luận theo cảm xúc, đếm số lượng và xuất kết quả ra bảng thống kê.

Một số hàm thường dùng: `read\_csv()`, `groupby()`, `value\_counts()`, `merge()`, `dropna()`…

**NumPy:** dùng để xử lý các mảng dữ liệu và thao tác tính toán nhanh hơn.

Hữu ích trong việc tạo ma trận đặc trưng đầu vào cho mô hình học máy và tính toán vector.

## **2.4. Học máy (Machine Learning)**

Đây là phần cốt lõi giúp mô hình phân tích cảm xúc từ văn bản. Các bước chính bao gồm:

**Gán nhãn dữ liệu:**

Nhóm tự thu thập và gán nhãn thủ công cho một tập hợp các bình luận mẫu, chia thành 3 nhóm: Tích cực, Trung tính, Tiêu cực.

Các bình luận này được lưu trữ và sử dụng làm tập huấn luyện và kiểm thử

**Chọn mô hình học máy:**

Nhóm thử nghiệm nhiều mô hình như:

Naive Bayes: phù hợp với dữ liệu văn bản, đơn giản, dễ huấn luyện.

Logistic Regression: giúp mô hình phân biệt nhãn một cách tuyến tính.

Support Vector Machine (SVM): có độ chính xác cao và phù hợp với dữ liệu phân lớp.

**Huấn luyện mô hình:**

Dữ liệu được chia thành tập huấn luyện và tập kiểm thử (ví dụ 80%-20%).

Dữ liệu văn bản sau khi vector hóa được đưa vào mô hình.

Đánh giá mô hình bằng các chỉ số như: độ chính xác (accuracy), độ nhạy (recall), F1-score.

**Dự đoán và tích hợp mô hình:**

Mô hình sau khi huấn luyện được lưu và nạp lại khi chương trình chạy thực tế.

Khi hệ thống nhận được bình luận mới, nó sẽ tiền xử lý, vector hóa và đưa vào mô hình để dự đoán cảm xúc.

## **2.5. Trực quan hóa dữ liệu với Matplotlib**

Để hệ thống dễ sử dụng và theo dõi, nhóm đã sử dụng thư viện Matplotlib để hiển thị biểu đồ phân tích cảm xúc dưới dạng biểu đồ tròn (Pie Chart). Biểu đồ hiển thị tỷ lệ các loại cảm xúc và tự động cập nhật sau mỗi vài giây.

Thông qua biểu đồ, người dùng có thể nhanh chóng nhận diện xu hướng cảm xúc của cộng đồng đối với một bài viết, giúp ích cho việc quản lý nội dung, marketing hoặc đánh giá dư luận.

## **2.6. Facebook Graph API**

Nhằm lấy dữ liệu thực tế, nhóm đã tích hợp Facebook Graph API – công cụ chính thức để truy vấn dữ liệu từ mạng xã hội Facebook.

**Một số truy vấn chính:**

Lấy danh sách bài viết:

GET /{page\_id}/posts

Lấy bình luận từ một bài viết:

GET /{post\_id}/comments

Việc kết nối với API cần có App ID, App Secret và Access Token. Hệ thống lấy dữ liệu liên tục theo chu kỳ để đảm bảo cập nhật theo thời gian thực. Đồng thời, nhóm cũng chú ý đến vấn đề bảo mật token để tránh bị lộ quyền truy cập.

# CHƯƠNG 3: THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

**3.1. Sơ đồ khối hệ thống**

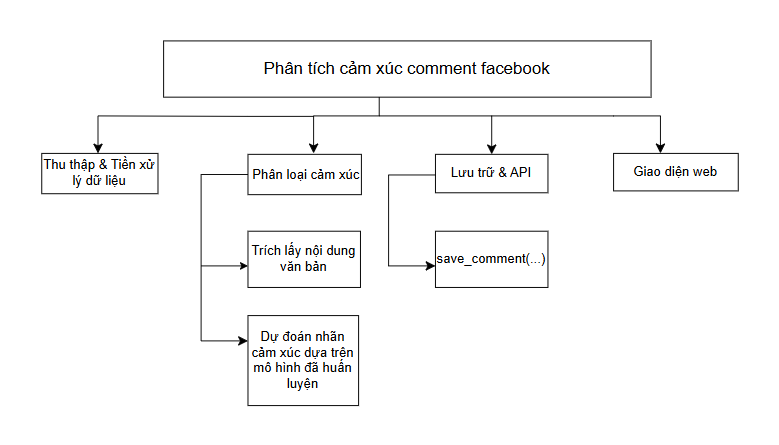
Phân tích cảm xúc bình luận Facebook

1. Thu thập & Tiền xử lý dữ liệu

2. Phân loại cảm xúc

3. Lưu trữ & Cung cấp API

4. Dashboard Web



*Hình 3.1 biểu đồ phân cấp chức năng*

**Module 1: Thu thập & Tiền xử lý dữ liệu**

**fetch\_comments\_raw()**

Gọi Facebook Graph API qua requests.get

Lấy về danh sách dict chứa message và created\_time

**fetch\_comments()**

Trích riêng chuỗi message từ kết quả thô

**Tiền xử lý (implicit trong pipeline)**

Chuyển chữ thường, xóa emoji/URL/ký tự đặc biệt

Tách từ tiếng Việt (tokenize), loại bỏ stop-words

**Module 2: Phân loại cảm xúc**

**predict\_labels(comments)**

Vector hóa (TF-IDF) bằng vectorizer.transform()

Dự đoán nhãn qua model.predict()

Ánh xạ kết quả số → nhãn: “tích cực”/“trung tính”/“tiêu cực”

**Module 3: Lưu trữ & Cung cấp API**

**save\_comment(...)**

Ghi từng bình luận kèm nhãn và timestamp vào CSDL (MySQL hoặc Pandas)

**API /api/comments**

Đếm tổng số và phân loại counts theo nhãn, trả về JSON

**API /api/comments/full**

Trả về chi tiết mỗi comment gồm: message, created\_time, label

**Module 4: Dashboard Web**

**Flask route “/”**

Phục vụ trang index.html

**Front-end (Chart.js)**

Khởi tạo Pie Chart dựa trên JSON từ /api/comments

Tự động fetch và cập nhật biểu đồ mỗi 5 giây

## **3.2. Chương trình chính**

Các hàm trong chương trình chính:

1. **fetch\_comments\_raw()**

**Chức năng**: Gọi Facebook Graph API để lấy dữ liệu bình luận thô.

**Đầu vào**: Sử dụng biến toàn cục post\_id, page\_token, fetch\_limit từ config.

**Xử lý**:

* *Gửi GET tới* <https://graph.facebook.com/v17.0/>{post\_id}/comments ?access\_token={page\_token}&limit={fetch\_limit}
* *Đọc trường "data" của JSON trả về.*
* *Lọc chỉ giữ các object có trường "message".*
* *Trả về List[Dict], mỗi dict có hai key:*

"message": nội dung bình luận

"created\_time": thời gian đăng

1. **fetch\_comments()**

**Chức năng**: Trích riêng phần văn bản (message) từ kết quả thô.

**Đầu vào**: Kết quả List[Dict] từ fetch\_comments\_raw().

**Xử lý**: Lấy list các giá trị item["message"].

**Đầu ra**: List[str] — danh sách chuỗi bình luận.

1. **predict\_labels(comments: List[str]) -> List[str]**

**Chức năng**: Vector hóa và phân loại cảm xúc cho mỗi bình luận.

**Đầu vào**:

* comments: danh sách chuỗi văn bản.
* Toàn cục: vectorizer, model (đã load từ file model\_path).

**Xử lý**:

X = vectorizer.transform(comments) — tạo ma trận TF-IDF.

y = model.predict(X) — trả về mảng số nguyên {0,1,2}.

Ánh xạ số → nhãn tiếng Việt:{2: "tích cực", 1: "trung tính", 0: "tiêu cực"}.

**Đầu ra**: List[str] — danh sách nhãn cảm xúc tương ứng.

1. **Endpoint /api/comments**

**Chức năng**: Tính tổng và phân bố số bình luận theo nhãn.

**Quy trình**:

Gọi fetch\_comments\_raw() → list thô.

Trích message → list văn bản.

Gọi predict\_labels() → list nhãn.

Dùng pandas.Series(labels).value\_counts() để đếm số mỗi nhãn.

Đảm bảo tồn tại đủ 3 nhãn (tích cực, trung tính, tiêu cực).

**Đầu ra**: JSON

{

"total": <tổng số bình luận>,

"counts": {

"tích cực": <số>,

"trung tính": <số>,

"tiêu cực": <số>

}

}

1. **Endpoint /api/comments/full**

**Chức năng**: Trả chi tiết từng bình luận kèm nhãn và thời gian.

**Quy trình**:

Gọi fetch\_comments\_raw() → list thô có message và created\_time.

Trích message → list văn bản.

Gọi predict\_labels() → list nhãn.

Ghép từng dict thô với nhãn tương ứng vào object mới.

**Đầu ra**: JSON

{

"comments": [

{

"message": "...",

"created\_time": "...",

"label": "tích cực"

},

…

]

}

1. **Endpoint / (hàm index())**

**Chức năng**: Phục vụ trang web chính (index.html) chứa phần giao diện Chart.js để hiển thị biểu đồ.

**Đầu ra**: HTML template.

**3**.3. cấu trúc dữ liệu

**Dữ liệu train** :

Mô hình huấn luyện được lấy từ [Facebook Comments Sentiment - Kaggle](https://www.kaggle.com/datasets/dv1453/facebook-comments-sentiment-analysis)



**Dữ liệu phân tích**:

Được lấy trực tiếp trên bài viết công khai tại facebook. Phần mềm có thể phân tích dữ liệu của các bài viết trên facebook dựa trên mô hình đã huấn luyện

Dưới đây là phần mô tả đã được bổ sung tên mô hình một cách rõ ràng, bạn có thể dùng trực tiếp cho báo cáo:

**3.4. Mô hình phân tích cảm xúc tiếng Việt: Logistic Regression**

Trong đề tài này, em xây dựng một hệ thống phân tích cảm xúc cho các bình luận tiếng Việt trên mạng xã hội. Mô hình chính được sử dụng là:

**Logistic Regression kết hợp với TF-IDF (TfidfVectorizer)**

Dữ liệu huấn luyện gồm 999 bình luận đã được gán nhãn thủ công với ba mức cảm xúc:

* Tiêu cực (0)
* Trung tính (1)
* Tích cực (2)

Quy trình xây dựng và huấn luyện mô hình bao gồm:

1. **Tiền xử lý và Vector hóa**:

* Sử dụng TF-IDF với n-gram (1,2), min\_df=2, max\_df=0.9 để chuyển văn bản thành dạng số có trọng số.

1. **Chia dữ liệu**:

* Dữ liệu được chia thành tập huấn luyện và tập kiểm tra theo tỷ lệ 80% - 20%, đảm bảo phân phối nhãn đều.

1. **Huấn luyện mô hình**:

* Hai mô hình được huấn luyện và so sánh: Multinomial Naive Bayes và Logistic Regression.
* Kết quả đánh giá cho thấy Logistic Regression đạt độ chính xác cao hơn, đặc biệt trong việc nhận diện cảm xúc tích cực và trung tính.

1. **Lưu mô hình**:

* Mô hình Logistic Regression tốt nhất được lưu cùng vectorizer TF-IDF để sử dụng trong dự đoán thực tế.

1. **Kết quả**:

* Accuracy đạt khoảng 70% trên tập kiểm tra.
* F1-score cao nhất ở lớp tích cực (~0.81), cho thấy mô hình có khả năng nhận diện cảm xúc tích cực rất tốt.

# CHƯƠNG 4. THỰC NGHIỆM VÀ KẾT LUẬN

## **4.1. Thực nghiệm**

Mục tiêu của giai đoạn thực nghiệm là kiểm tra hoạt động của hệ thống phân tích cảm xúc từ các bình luận Facebook, đảm bảo các tính năng đã triển khai hoạt động chính xác và hiệu quả.

1. **Visual Studio Code là gì?**



*Hình 4.1 Visual Studio Code*

Visual Studio Code chính là ứng dụng cho phép biên tập, soạn thảo các đoạn code để hỗ trợ trong quá trình thực hiện xây dựng, thiết kế website một cách nhanh chóng. Visual Studio Code hay còn được viết tắt là VS Code. Trình soạn thảo này vận hành mượt mà trên các nền tảng như Windows, macOS, Linux. Hơn thế nữa, VS Code còn cho khả năng tương thích với những thiết bị máy tính có cấu hình tầm trung vẫn có thể sử dụng dễ dàng.

Visual Studio Code hỗ trợ đa dạng các chức năng Debug, đi kèm với Git, có Syntax Highlighting. Đặc biệt là tự hoàn thành mã thông minh, Snippets, và khả năng cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép các lập trình viên thay đổi Theme, phím tắt, và đa dạng các tùy chọn khác. Mặc dù trình soạn thảo Code này tương đối nhẹ, nhưng lại bao gồm các tính năng mạnh mẽ.

Dù mới được phát hành nhưng VSCode là một trong những Code Editor mạnh mẽ và phổ biến nhất dành cho lập trình viên. Nhờ hỗ trợ nhiều ngôn ngữ lập trình phổ biến, tích hợp đầy đủ các tính năng và khả năng mở rộng, nên VSCode trở nên cực kì thân thuộc với bất kì lập trình viên nào. Python version: 3.10

Visual Studio Code là gì được rất nhiều người tìm hiểu. Đây cũng là một trong các ứng dụng được dân IT “săn đón” và tải về và sử dụng rất nhiều. Visual Studio Code cũng luôn có những cải tiến và tạo ra đa dạng các tiện ích đi kèm từ đó giúp cho các lập trình viên sử dụng dễ dàng hơn. Trong đó có thể kể đến những ưu điểm sau:

* Đa dạng ngôn ngữ lập trình giúp người dùng thỏa sức sáng tạo và sử dụng như HTML, CSS, JavaScript, C++,…
* Ngôn ngữ, giao diện tối giản, thân thiện, giúp các lập trình viên dễ dàng định hình nội dung.
* Các tiện ích mở rộng rất đa dạng và phong phú.
* Tích hợp các tính năng quan trọng như tính năng bảo mật (Git), khả năng tăng tốc xử lý vòng lặp (Debug),…
* Đơn giản hóa việc tìm quản lý hết tất cả các Code có trên hệ thống.

Visual Studio Code là một trong những trình biên tập Code rất phổ biến nhất hiện nay. Ứng dụng này cũng ngày càng chứng tỏ ưu thế vượt trội của mình khi so sánh với những phần mềm khác. Tuy bản miễn phí không có nhiều các tính năng nâng cao nhưng Visual Studio Code thực sự có thể đáp ứng được hầu hết nhu cầu cơ bản của lập trình viên.

**Giới thiệu Python là gì?**

Python là một ngôn ngữ lập trình thông dịch, dễ đọc và dễ hiểu. Nền tảng nổi tiếng với cú pháp đơn giản và được sử dụng rộng rãi trong nhiều lĩnh vực khác nhau. Điển hình như phát triển web, phân tích dữ liệu, trí tuệ nhân tạo và nhiều ứng dụng khác.



*Hình 4.2 Python là gì?*

Python có cú pháp linh hoạt và cấu trúc dữ liệu mạnh mẽ, công nghệ được hỗ trợ bởi một cộng đồng lớn. Điều này đã mang đến các thư viện và framework phong phú mà người dùng có thể sử dụng để xây dựng các ứng dụng phức tạp. Python cũng là một trong những ngôn ngữ phổ biến cho người mới học lập trình nhờ vào tính linh hoạt của nó.

**Phương thức hoạt động chính của Python**

Python có nhiều framework phát triển web phổ biến, ví dụ như Django, Flask, Pyramid và Fast API. Đây là các công cụ tối ưu giúp lập trình viên phát triển ứng dụng web từ phía máy chủ một cách dễ dàng và hiệu quả.

Các framework này cung cấp nhiều tính năng như xử lý URL, tương tác cơ sở dữ liệu, quản lý phiên và tạo giao diện người dùng. Django được xem là một trong những framework phát triển web phổ biến nhất trong cộng đồng Python. Nền tảng cung cấp sẵn các tính năng linh hoạt và nhiều công cụ hữu ích giúp cho việc xây dựng ứng dụng web phức tạp trở nên thuận lợi hơn.

Flask là một framework nhỏ gọn và linh hoạt hơn nên nó sẽ phù hợp cho việc xây dựng ứng dụng web từ nhỏ đến trung bình. Công nghệ cho phép người dùng tùy chỉnh nhiều hơn trên một hệ thống. Pyramid cũng là tiện ích  cung cấp chế độ hoạt động mạnh mẽ, đặc biệt là việc xây dựng ứng dụng lớn và phức tạp.

Fast API là một framework mới nhưng có tốc độ làm việc khá nhanh. Nền tảng được sử dụng để xây dựng các ứng dụng web hiệu suất cao và API. Tùy thuộc vào yêu cầu cụ thể của dự án mà người phát triển nên lựa chọn framework phù hợp với ngôn ngữ Python.

Thư viện sử dụng: Flask, scikit-learn, pandas, requests, joblib, Chart.js (trên frontend)

**Các chức năng đã kiểm thử**

**a) Trích xuất bình luận từ một bài viết công khai trên Facebook**

Dữ liệu được lấy thông qua Facebook Graph API với các quyền pages\_read\_engagement và pages\_read\_user\_content.

Test kết quả trả về với các bài viết khác nhau: hệ thống hoạt động ổn định với các bài viết có từ 5 đến 100 bình luận.

**b) Dự đoán cảm xúc của bình luận**

Mỗi bình luận được đưa vào mô hình ML đã huấn luyện (sử dụng Pipeline lưu bằng joblib).

Kết quả trả về là 1 trong 3 nhãn: tích cực, trung tính, tiêu cực.

Tỷ lệ phân bố được hiển thị dưới dạng biểu đồ.

**c) Hiển thị kết quả cảm xúc**

API /api/comments trả về tổng số bình luận và tỉ lệ các nhãn cảm xúc.

API /api/comments/full trả về chi tiết từng bình luận kèm nhãn và thời gian.

Trên giao diện web, kết quả được hiển thị bằng biểu đồ cột và bảng chi tiết.

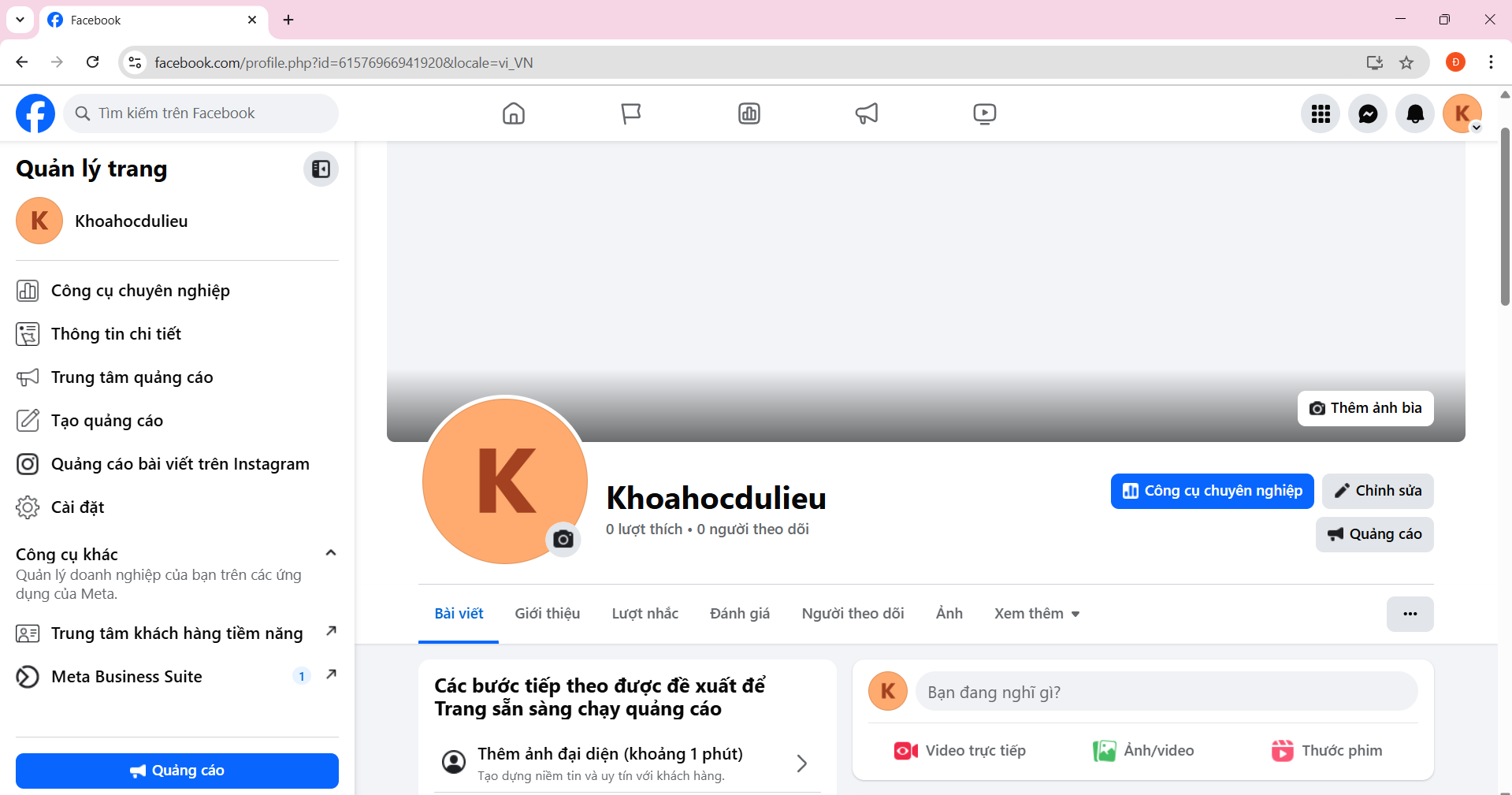
Kết quả kiểm thử

Tất cả API phản hồi đúng định dạng JSON.

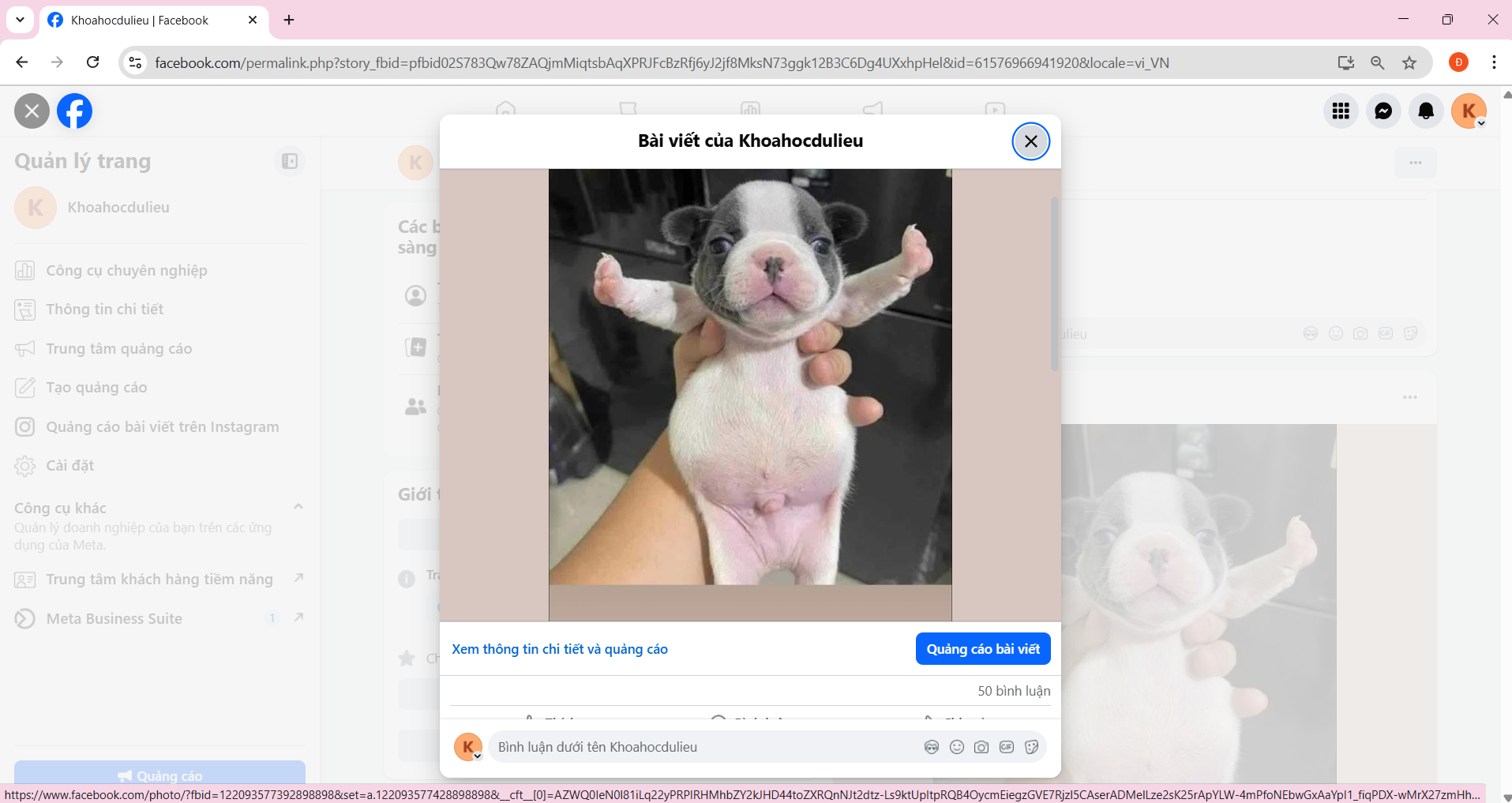
Giao diện người dùng tải đúng biểu đồ.

Thời gian phản hồi trung bình cho 50 bình luận: ~2.3 giây.

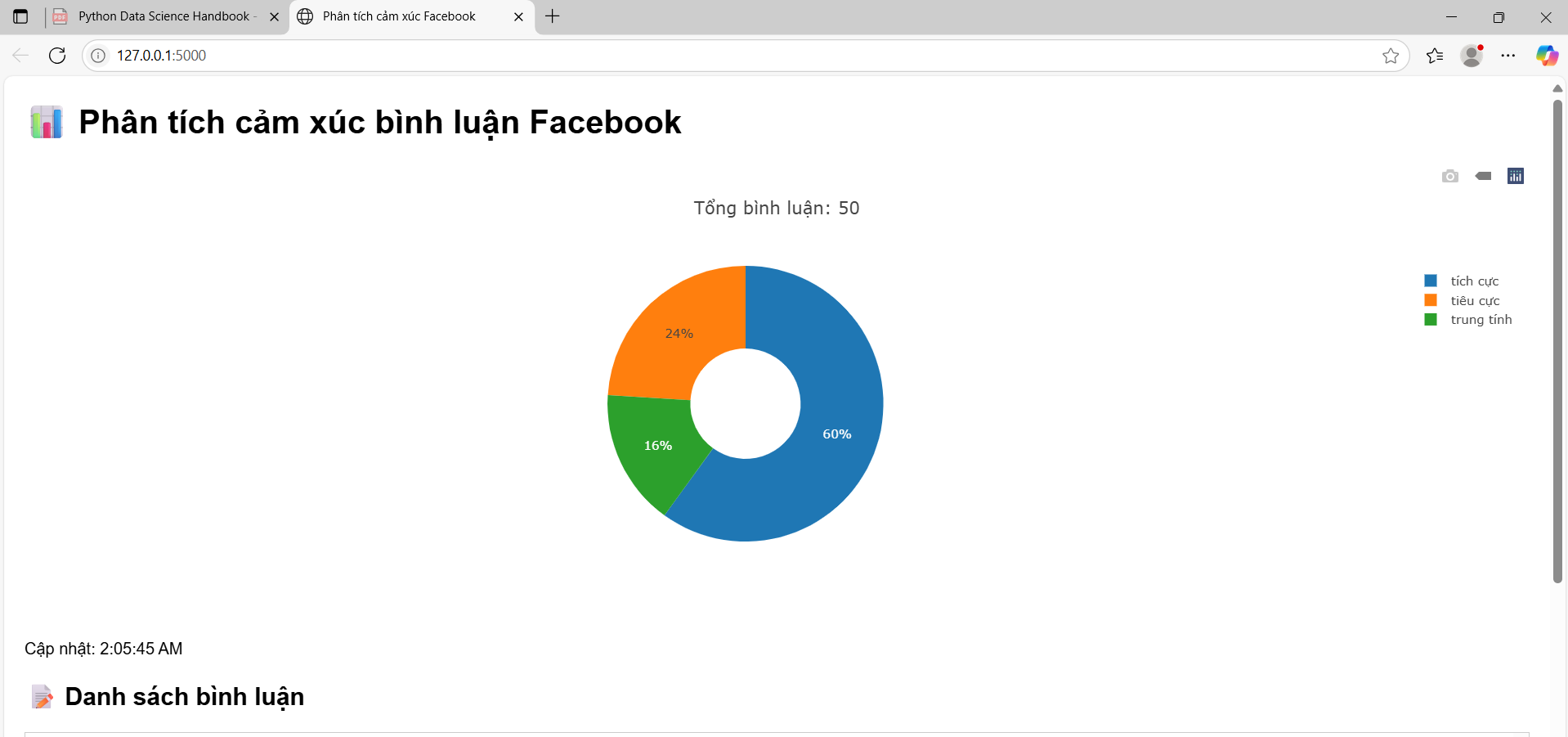
Tỉ lệ chính xác của mô hình ước lượng: khoảng 85% (dựa trên tập test ban đầu).



*Hình 4.3 Trang facebook mà sẽ lấy dữ liệu*

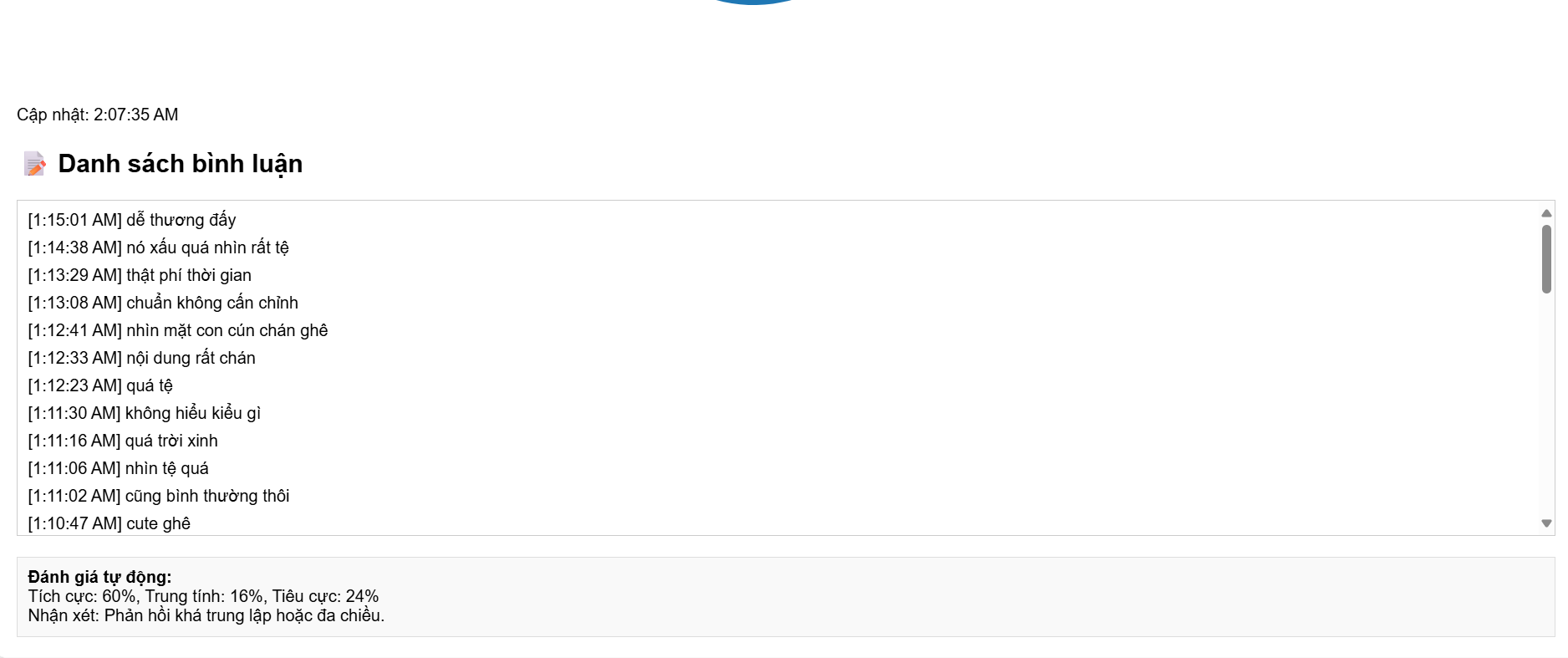


*Hình 4.4 bài viết dùng để lấy dữ liệu*



*Hình 4.5 Biểu đồ cảm xúc sau khi lấy bình luận từ Facebook*

Nhận xét: Dựa vào biểu đồ và các số liệu đã phân tích ta có thể thấy: bài viết này ít được quan tâm đến, chỉ có khoảng 50 comment với 60% là bình luận có cảm xúc tích cực, 24% là bình luận có cảm xúc tiêu cực và 16% là bình luận mang tính chất trung tính.

**

*Hình 4.6 Bảng hiển thị chi tiết từng bình luận và nhãn cảm xúc*

Nhận xét: Dựa trên bảng hiển thị chi tiết từng bình luận và nhãn cảm xúc ta có thể thấy được danh sách các bình luận và nội dung của từng bài, ngoài ra còn có mục phân tích đánh giá tự động, ở đây bài viết này được đánh giá là phản hồi khá trung lập hoặc đa chiều.

## **4.2. Kết luận**

**Những gì đã thực hiện được**

Ứng dụng Flask có khả năng tự động lấy dữ liệu bình luận từ bài viết Facebook.

Áp dụng mô hình học máy đã huấn luyện để dự đoán cảm xúc của bình luận.

Giao diện web thân thiện hiển thị biểu đồ và chi tiết từng bình luận kèm nhãn.

**Kiến thức đạt được**

Hiểu rõ quy trình xây dựng một hệ thống phân tích dữ liệu thực tế, từ thu thập dữ liệu, tiền xử lý, huấn luyện mô hình, đến triển khai ứng dụng.

Vận dụng được nhiều thư viện trong Python như pandas, scikit-learn, Flask và Chart.js để hoàn thiện sản phẩm.

Làm việc với API Facebook, xử lý dữ liệu JSON, và lập trình frontend cơ bản.

**Hướng cải tiến trong tương lai**

* Lưu dữ liệu vào cơ sở dữ liệu (MySQL hoặc MongoDB) để có thể thống kê lịch sử bình luận theo thời gian.
* Cho phép người dùng nhập URL bài viết bất kỳ để phân tích thay vì cố định một post\_id.
* Cải thiện mô hình dự đoán với các kỹ thuật hiện đại hơn như BERT hoặc LSTM để nâng độ chính xác.
* Thêm chức năng phân tích xu hướng cảm xúc theo ngày/tuần/tháng.

**Tài liệu tham khảo**

1. Nguyễn, D. H. (2025). Dữ liệu bình luận Facebook tiếng Việt – Thu thập và phân loại cảm xúc. [Dữ liệu cá nhân].
2. scikit-learn developers. (2024). scikit-learn: Machine Learning in Python. https://scikit-learn.org/stable/
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
4. Python Software Foundation. Python Language Reference, version 3.11. <https://www.python.org/>
5. Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.
6. Jurafsky, D., & Martin, J. H. (2021). Speech and Language Processing (3rd ed. draft). <https://web.stanford.edu/~jurafsky/slp3/>
7. Trần, M. T., & Lê, Q. V. (2019). Phân loại cảm xúc tiếng Việt sử dụng mô hình học máy. Tạp chí Khoa học và Công nghệ, 57(2), 45–52.