TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Nguyễn Tiến Nhật - Bùi Đình Bảo

MÔ HÌNH PHÁT HIỆN ẢO GIÁC CỦA MÔ HÌNH NGÔN NGỮ LỚN TRONG NGỮ CẢNH DỊCH VỤ CÔNG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

Tp. Hồ Chí Minh, tháng 07/2025

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

Nguyễn Tiến Nhật - 21120108
Bùi Đình Bảo - 21120201

MÔ HÌNH PHÁT HIỆN ẢO GIÁC CỦA MÔ HÌNH NGÔN NGỮ LỚN TRONG NGỮ CẢNH DỊCH VỤ CÔNG

KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN
CHƯƠNG TRÌNH CHÍNH QUY

GIÁO VIÊN HƯỚNG DẪN
TS. Nguyễn Tiến Huy
ThS. Nguyễn Trần Duy Minh

Tp. Hồ Chí Minh, tháng 07/2025

6

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

BẢN NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(HƯỚNG NGHIÊN CỨU)

Tên đề tài: Mô Hình Phát Hiện Ảo Giác Của Mô Hình Ngôn Ngữ Lớn Trong Ngữ Cảnh Dịch Vụ Công

Sinh viên thực hiện : Nguyễn Tiến Nhật - 21120108
Bùi Đình Bảo - 21120201

Giảng viên hướng dẫn: TS. Nguyễn Tiến Huy – ThS. Nguyễn Trần Duy Minh

1. **Chủ đề và ý tưởng nghiên cứu:**

Hiện tượng ảo giác (hallucination) của các mô hình ngôn ngữ lớn (LLMs) là vấn đề "nóng", đặc biệt trong ngữ cảnh dịch vụ công, nơi độ chính xác và khả năng kiểm chứng thông tin rất quan trọng. Nhóm tập trung xây dựng bộ dữ liệu tiếng Việt chuyên biệt cho phát hiện ảo giác trong dịch vụ công—một lĩnh vực vẫn còn rất thiếu tập dữ liệu đánh giá.

2. **Phương pháp nghiên cứu:**

Dựa trên phương pháp HaluEval được để xuất ở tiếng Anh, đề tài đề xuất pipeline ba giai đoạn: (i) thu thập dữ liệu, (ii) tiền xử lý & kiểm tra thủ công, (iii) sinh và chú thích phản hồi ảo giác. Bộ dữ liệu được đánh giá trên các mô hình ngôn ngữ lớn nhằm đánh giá mức độ ảo giác của các mô hình cho ngữ cảnh dịch vụ công tiếng Việt.

3. **Đóng góp Khoa học và thực tiễn:**

Khóa luận xây dựng tập dữ liệu ảo giác với 3717 mẫu dữ liệu thuộc nhiều chủ đề được thu thập từ cổng dịch vụ công quốc gia. Dữ liệu cũng chia ra 4 dạng ảo giác phổ biến nhằm đánh giá chi tiết các loại ảo giác mà mô hình gặp phải. Bộ dữ liệu và quy trình tạo dữ liệu này là nền tảng để phát triển trợ lý ảo hành chính an toàn, giảm rủi ro thông tin sai lệch cho người dân.

4. **Quá trình thực hiện và quản lý dự án:**

Nhóm sinh viên làm việc nghiêm túc và có trách nhiệm.

5. **Báo cáo viết:**

Báo cáo được chia thành 5 chương dài 72 trang với bố cục hợp lý, và mạch lạc.

6. **Trình bày trước hội đồng:**

Tốt

7. **Công bố khoa học/ ứng dụng thực tế:**

Không có

Đánh giá xếp loại: Đạt yêu cầu của một khóa luận đại học

TP.HCM, ngày …. tháng …. năm …….

Giảng viên hướng dẫn

(*Ký và ghi rõ họ tên*)

Nguyễn Tiến Huy **Nguyễn Trần Duy Minh**

b

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

BẢN NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP

(HƯỚNG NGHIÊN CỨU)

Tên đề tài : MÔ HÌNH PHÁT HIỆN ẢO GIÁC CỦA MÔ HÌNH NGÔN NGỮ LỚN TRONG NGỮ CẢNH DỊCH VỤ CÔNG

Sinh viên thực hiện : 21120108 – Nguyễn Tiến Nhật

21120201 – Bùi Đình Bảo

Giảng viên phản biện: TS. Nguyễn Ngọc Thảo

1. Chủ đề và ý tưởng nghiên cứu:

Khóa luận tập trung vào hiện tượng "ảo giác" (hallucination) của các mô hình ngôn ngữ lớn (LLM) – một vấn đề quan trọng nhưng còn ít được nghiên cứu trong ngữ cảnh tiếng Việt, đặc biệt là dịch vụ công. Nhóm tác giả đề xuất xây dựng một mô hình phát hiện ảo giác dựa trên kỹ thuật tạo dữ liệu có giám sát kết hợp với đánh giá nhiều mô hình ngôn ngữ lớn hiện nay. Ý tưởng mang tính thời sự, có giá trị ứng dụng trong bối cảnh Việt Nam đang thúc đẩy chuyển đổi số.

2. Phương pháp nghiên cứu:

Khóa luận xây dựng một quy trình phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công thông qua năm bước chính. Đầu tiên, nhóm thu thập hơn dữ liệu từ Cổng Dịch vụ công Quốc gia, gồm câu hỏi, câu trả lời và thủ tục hành chính liên quan. Sau đó, dữ liệu được tiền xử lý để loại bỏ trùng lặp, chuẩn hóa văn bản và đảm bảo tính đầy đủ thông tin. Tiếp theo, nhóm thiết kế hệ thống prompt theo bốn loại lỗi phổ biến (hiểu sai ngữ cảnh, mâu thuẫn tri thức, quá chung/chi tiết, suy luận sai) để tạo ra các phản hồi ảo giác từ mô hình LLM. Các phản hồi được gán nhãn thủ công bởi người có chuyên môn để tạo thành bộ dữ liệu tiêu chuẩn. Cuối cùng, nhóm đánh giá khả năng phát hiện ảo giác của nhiều mô hình ngôn ngữ lớn mã nguồn mở và đóng (GPT-4o, WizardLM-2, Claude, DeepSeek...) thông qua các độ đo như accuracy, phân tích theo lĩnh vực và pattern lỗi. Phương pháp có tính hệ thống, thực nghiệm rõ ràng, giúp xác định mô hình phù hợp cho ứng dụng trợ lý ảo trong dịch vụ công.

3. Đóng góp Khoa học và thực tiễn:

Khóa luận đóng góp bộ dữ liệu tiếng Việt đầu tiên về phát hiện ảo giác trong ngữ cảnh dịch vụ công – VietPS-Hallu, với hơn 3.700 mẫu gán nhãn. Ngoài ra, khóa luận đề xuất quy trình xây dựng dữ liệu có thể mở rộng và đánh giá thực nghiệm trên nhiều mô hình để lựa chọn phương án hiệu quả. Các kết quả có thể ứng dụng trong việc phát triển trợ lý ảo hành chính công đáng tin cậy và có thể mở rộng sang các lĩnh vực khác.

4. Báo cáo viết:

Báo cáo viết gồm 74 trang chia thành 05 chương và 04 phụ lục, trình bày mạch lạc, rõ ràng, có cấu trúc hợp lý, và ngôn ngữ học thuật phù hợp. Các bảng số liệu, hình ảnh minh họa được thiết lập có mục đích rõ ràng, được chú thích đầy đủ.

5. Trình bày trước hội đồng:

Nhóm sinh viên trình bày tốt, nắm vững nội dung và trả lời các câu hỏi phản biện một cách rõ ràng và đầy đủ.

6. Công bố khoa học/ ứng dụng thực tế:

Chưa có.

Đánh giá xếp loại: Đạt yêu cầu của Khóa luận tốt nghiệp bậc Đại học Chương trình Chuẩn.

TP.HCM, ngày 01 tháng 8 năm 2025

Giảng viên phản biện

Nguyễn Ngọc Thảo

Lời cảm ơn

Trước tiên, chúng tôi xin chân thành cảm ơn gia đình luôn tin tưởng vào chúng tôi. Sự động viên không ngừng nghỉ của họ là nguồn động lực của chúng tôi trên con đường học thuật và sự nghiệp nghiên cứu.

Tiếp theo, chúng tôi xin bày tỏ lòng biết ơn tới TS. Nguyễn Tiến Huy, TS. Lê Thanh Tùng và ThS. Nguyễn Trần Duy Minh vì sự hướng dẫn tận tình của các thầy trong suốt quá trình chúng tôi thực hiện luận văn tốt nghiệp này. Các thầy luôn tận tâm chỉ dẫn chúng tôi vượt qua những khó khăn và thử thách trong nghiên cứu khoa học. Ngoài ra, chúng tôi đã học được rất nhiều điều quý báu từ các thầy, giúp chúng tôi hình thành phong cách làm việc hiệu quả và chuyên nghiệp hơn. Những điều đó chính là chất liệu quan trọng cho các ý tưởng nghiên cứu của chúng tôi, và giúp chúng tôi vượt qua nhiều vấn đề khó khăn trong cả dự án và cuộc sống.

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

ĐỀ CƯƠNG KHÓA LUẬN TỐT NGHIỆP

MÔ HÌNH PHÁT HIỆN ẢO GIÁC CỦA MÔ HÌNH NGÔN NGỮ LỚN TRONG NGỮ CẢNH DỊCH VỤ CÔNG

1 THÔNG TIN CHUNG

Giảng viên hướng dẫn:

- TS. Nguyễn Tiến Huy (Khoa Công nghệ Thông tin)
- ThS. Nguyễn Trần Duy Minh (Khoa Công nghệ Thông tin)

Nhóm sinh viên thực hiện:

1. Nguyễn Tiến Nhật (MSSV: 21120108)
2. Bùi Đình Bảo (MSSV: 21120201)

Loại đề tài: Nghiên cứu

Thời gian thực hiện: Từ *01/2025* đến *07/2025*

2 NỘI DUNG THỰC HIỆN

2.1 Giới thiệu tổng quát

Mô hình ngôn ngữ lớn (Large Language Model - LLM) là các mô hình học sâu rất lớn, được đào tạo trước (pre-trained) dựa trên một lượng dữ liệu khổng lồ. Một trong những thành công quan trọng nhất của LLM là nó có khả năng hiểu và sinh ngôn ngữ tự nhiên như con người, mô hình ngôn ngữ lớn có thể thực hiện các tác vụ hoàn toàn khác nhau, ví dụ như trả lời câu hỏi, tóm tắt tài liệu, dịch ngôn ngữ và hoàn thành câu. Ngày nay, mô hình ngôn ngữ lớn đã và đang được ứng dụng rộng rãi trên nhiều lĩnh vực khác nhau như giáo dục, y tế, thương mại, truyền thông, quảng cáo,... Xét về khía cạnh dịch vụ công, mô hình ngôn ngữ lớn cũng cho thấy được tiềm năng vì nó có thể tạo ra xử lý và tạo sinh văn bản, từ đó có thể hỗ trợ được người dân nhanh chóng hơn trong các vấn đề về giấy tờ, hồ sơ liên quan, các bước thực hiện thủ tục hành chính.

Một trong những hạn chế lớn nhất của mô hình ngôn ngữ lớn đó là nó có thể tạo ra văn bản không chính xác, vô nghĩa hoặc không dựa trên dữ liệu thực tế. Hiện tượng này còn được gọi là ảo giác (hallucination) của mô hình ngôn ngữ lớn. Nguyên nhân của vấn đề này xuất phát từ nhiều tác động khác nhau, có thể là từ dữ liệu được đưa vào không đúng đắn hay là từ quá trình huấn luyện mô hình không thể kiểm soát được. Quay trở lại với ngữ cảnh dịch vụ công thì hiện tượng ảo giác này lại tạo ra nhiều thách thức hơn nữa khi tính chính xác và độ tin cậy đầu ra (output) của mô hình là vô cùng quan trọng.

Chính vì thế, sau quá trình tìm hiểu, nhóm em xây dựng một mô hình dùng để phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công. Nền tảng phía sau mô hình này chính là một khung phương pháp hoàn chỉnh từ bước chuẩn bị dữ liệu cho đến khi đánh giá hiệu suất của các mô hình ngôn ngữ lớn khác nhau. Ý tưởng cho phương pháp này dựa trên kỹ thuật tạo lời nhắc (prompt engineering) từ đó ta sẽ tinh chỉnh và tạo ra output phù hợp cho tác vụ cần xử lý.

Những kết quả đạt được từ mô hình phát hiện ảo giác trên dự kiến sẽ bao gồm một bộ dữ liệu chất lượng cao, đảm bảo được khả năng đánh giá khách quan về hiệu suất đầu ra của các mô hình ngôn ngữ lớn, từ đó góp phần giúp cho quá trình tinh chỉnh mô hình đạt được hiệu quả tốt nhất. Đặc biệt, ngôn ngữ của bộ dữ liệu sẽ hoàn toàn là tiếng Việt, điều này giúp cho tiêu chuẩn đánh giá trở nên chuyên dụng và thiết thực hơn trong ngữ cảnh dịch vụ công ở nước ta. Áp dụng vào thực tiễn, nhóm em hi vọng mô hình có thể giúp khắc phục được hiện tượng ảo giác của mô hình ngôn ngữ lớn, với mục đích khiến cho câu trả lời trở nên minh bạch và đúng đắn hơn. Dựa vào đó, các trợ lý ảo trí tuệ nhân tạo dịch vụ công có thể được xây dựng để hỗ trợ người dân kịp thời và tốt hơn, nhằm nâng cao đời sống cộng đồng về lâu dài.

2.2 Mục tiêu nghiên cứu

Đứng trước sự bùng nổ của các mô hình ngôn ngữ lớn, việc tận dụng sức mạnh của nó là một điều cần thiết nhằm tự động hóa các tác vụ, mang đến khả năng phản hồi tức thời cho những khó khăn của con người. Tuy nhiên, việc áp dụng công nghệ này vẫn đi kèm theo nhiều thách thức đặt ra, một trong số đó là hiện tượng ảo giác của mô hình tạo sinh ngôn ngữ, khiến cho câu trả lời đầu ra trở nên sai lệch, vô lý hoặc xa rời thực tế. Điều này là vấn đề rất nhạy cảm trong ngữ cảnh dịch vụ công khi những vấn đề pháp lý và chứng thực thông tin là không thể bỏ qua. Đồng thời, hầu hết các mô hình ngôn ngữ lớn hiện chỉ đang phục vụ cho tiếng Anh là chủ yếu, trong khi đó hỗ trợ cho tiếng Việt cụ thể hơn là các mô hình huấn luyện, đánh giá và các bộ dữ liệu tiêu chuẩn cho tiếng Việt vẫn còn đang rất hiếm trong hàng loạt các lĩnh vực nói chung, chứ không riêng gì ngữ cảnh dịch vụ công. Vì vậy, mô hình phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công được đề xuất nhằm khắc phục những nhược điểm trên.

Với đề tài nghiên cứu này, nhóm em mong muốn đem lại:

- Một quy trình hoàn chỉnh dùng để đánh giá ảo giác của mô hình ngôn ngữ

lớn trong ngữ cảnh dịch vụ công, hỗ trợ tiếng Việt.

- Một bộ dữ liệu phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công, từ đó giúp tinh chỉnh mô hình đồng thời có thể làm một tiêu chuẩn (benchmark) phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công, hỗ trợ tiếng Việt.
- Một bảng thống kê so sánh hiệu suất của các mô hình ngôn ngữ lớn phổ biến hiện nay trong ngữ cảnh dịch vụ công, cụ thể hơn là các chủ đề trong ngữ cảnh dịch vụ công, từ đó đưa ra mô hình phù hợp nhất cho tác vụ này.

Từ những kết quả trên, những ý nghĩa thực tiễn và ảnh hưởng tích cực của mô hình không chỉ dừng lại ở việc có thể khắc phục được hiện tượng ảo giác của mô hình ngôn ngữ lớn trong bối cảnh dịch vụ công, mà còn hứa hẹn có thể đề ra phương pháp chung để cải thiện hiệu suất của mô hình tạo sinh ngôn ngữ cho những lĩnh vực khác. Đi đôi với việc giảm thiểu hiện tượng ảo giác của mô hình ngôn ngữ lớn, ta còn có thể tinh chỉnh những mô hình này từ đó tạo ra những trợ lý ảo trí tuệ nhân tạo hỗ trợ con người tốt hơn về những vấn đề xung quanh dịch vụ công.

2.3 Phạm vi của đề tài

Đối với phạm vi của đề tài, các yếu tố liên quan là:

- Đối tượng nghiên cứu của đề tài bao gồm các mô hình ngôn ngữ lớn (LLM) như GPT4o-mini, DeepSeek, Gemini hoặc các mô hình tương tự, đặc biệt là khi chúng được áp dụng vào các hệ thống dịch vụ công.
- Thực thể liên quan bao gồm:
  - **Người dùng dịch vụ công**: Là những người trực tiếp tương tác với các hệ thống AI trong các dịch vụ công như y tế, hành chính, bảo hiểm xã hội,...

- **Cơ quan nhà nước và các dịch vụ công**: Các tổ chức, cơ quan nhà nước triển khai các mô hình ngôn ngữ lớn để cung cấp các dịch vụ tự động hoặc trợ giúp thông qua các chatbot, trợ lý ảo.
- **Mô hình ngôn ngữ lớn (LLM)**: Các mô hình AI được sử dụng để xử lý, tạo ra phản hồi, và hỗ trợ người dùng trong các dịch vụ công, đặc biệt là các mô hình có khả năng gây ra hiện tượng ảo giác.

• Về tập dữ liệu, có 2 tập dữ liệu chính là:

- **Tập dữ liệu lấy từ trang web chính thống**: hơn 9000 mẫu gồm câu hỏi thường gặp, câu trả lời, bộ ngành và thủ tục hành chính được lấy từ trang 'dichvucong.gov.vn' của chính phủ Việt Nam để đảm bảo tính đúng đắn của tập dữ liệu. Sau khi trải qua bước tiền xử lý dữ liệu thì tập dữ liệu sau cùng còn gần 7000 mẫu.
- **Tập dữ liệu câu trả lời ảo giác**: gồm gần 7000 mẫu câu trả lời ảo giác, được sinh ra từ mô hình ngôn ngữ lớn (LLM) dựa trên tập dữ liệu câu hỏi thường gặp.

2.4 Cách tiếp cận dự kiến

2.4.1 Các công trình liên quan

Các nghiên cứu trước đây đã từng đề cập đến vấn đề phát hiện ảo giác của các mô hình ngôn ngữ lớn nói chung bao gồm các tiêu chuẩn (benchmark) hay cụ thể hơn là các bộ dữ liệu dùng để phục vụ như một nền tảng giúp hiển thị được các ảo giác trong mô hình ngôn ngữ lớn như:

• **BEGIN**, một bộ dữ liệu phân loại các câu trả lời của hệ thống đối thoại thành 3 loại là: fully attributable (tạm dịch là hoàn toàn được hỗ trợ bởi tri thức), not fully attributable (tạm dịch là không hoàn toàn được hỗ trợ bởi tri thức) và generic (câu trả lời chung chung) [1].

- **AIS**, một bộ dữ liệu đánh giá liệu các tài liệu nguồn có hỗ trợ đầu ra của các mô hình tạo sinh văn bản hay không [2].
- **SelfCheckGPT-Wikibio**, một bộ dữ liệu ở cấp độ câu được tạo ra bằng cách tổng hợp các bài viết từ Wikipedia với GPT-3, được chú thích thủ công về tính thực tế, gây ra thách thức cho việc phát hiện ảo giác về tiểu sử cá nhân [3].
- **FELM**, một bộ dữ liệu được chú thích về tính thực tế trên nhiều lĩnh vực khác nhau bao gồm kiến thức thế giới, khoa học và toán học [4].
- **HaluEval**, một bộ dữ liệu kết hợp tạo sinh tự động với chú thích của con người để đánh giá khả năng nhận diện ảo giác của các mô hình ngôn ngữ lớn [5].

Các nghiên cứu trên đã chứng minh được tính hiệu quả khi có thể đánh giá được các mô hình ngôn ngữ lớn khác nhau trong việc tạo ra ảo giác trên một số những lĩnh vực nhất định hay trải dài trên nhiều khía cạnh khác nhau.

Xét đến ngữ cảnh dịch vụ công, mô hình ngôn ngữ lớn cũng đã được áp dụng trong thực tế, **trợ lý ảo dịch vụ công** đã được xây dựng để hỗ trợ trên 15 chủ đề thiết yếu nhất mà người dân gặp phải khó khăn [6].

2.4.2 Điểm hạn chế tương ứng

Dù đã có nhiều tiêu chuẩn (bộ dữ liệu) trước đó nói về việc phát hiện ảo giác của mô hình ngôn ngữ lớn, các nghiên cứu chỉ mới dừng lại ở việc đánh giá trên ngữ cảnh tổng thể, chung chung trên toàn bộ mô hình ngôn ngữ lớn. Các lĩnh vực liên quan như y tế, giáo dục thường được chú trọng nhiều hơn trong khi lĩnh vực dịch vụ công vẫn còn khá sơ khai.

Các bộ dữ liệu trước đây chỉ được tập trung xây dựng cho tiếng Anh và một số ngoại ngữ phổ biến khác. Chính vì thế, việc sử dụng những tiêu chuẩn đánh giá này chỉ tối ưu nhất khi áp dụng cho tiếng Anh mà thôi.

Ở Việt Nam, mô hình ngôn ngữ lớn vẫn đang trong quá trình phát triển và được ứng dụng rộng rãi trên nhiều lĩnh vực khác nhau. Trong bối cảnh dịch vụ công, mô hình ngôn ngữ lớn cũng được áp dụng nhưng vẫn chưa được phổ biến với người dân. Các trợ lý ảo dịch vụ công này chỉ việc đưa ra câu trả lời dựa trên tài liệu thủ tục hành chính nhất định mà không hề quan tâm đến khả năng tạo sinh ảo giác của nó.

Chính vì thế, mô hình phát hiện ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công được đề xuất để đánh giá chất lượng của các trợ lý ảo dịch vụ công, đồng thời có thể lấy đó làm cơ sở để cải tiến mô hình trong tương lai.

2.4.3 Phương pháp chính

Quá trình nghiên cứu bắt đầu bằng việc thu thập dữ liệu từ Cổng dịch vụ công quốc gia Việt Nam, bao gồm các câu hỏi thường gặp về dịch vụ công trong nhiều lĩnh vực khác nhau. Các câu trả lời được tham chiếu từ các tài liệu thủ tục hành chính chính thống, giúp đảm bảo tính chính xác. Bộ dữ liệu thô được xây dựng bao gồm câu hỏi, lĩnh vực liên quan, câu trả lời chính xác và tài liệu tham chiếu để làm cơ sở cho việc xử lý tiếp theo.

Dữ liệu sau khi thu thập được tiến hành tiền xử lý, bao gồm xử lý chuỗi để chuẩn hóa văn bản, loại bỏ các trùng lặp về câu hỏi và câu trả lời, đồng thời bổ sung thông tin cho các câu hỏi thiếu thủ tục hành chính. Sau đó, dựa theo bài báo HaluEval, một khung prompt được thiết kế theo cấu trúc in-context learning nhằm tạo ra các ảo giác có kiểm soát, giúp đánh giá khả năng tạo ảo giác của mô hình ngôn ngữ lớn.

Cuối cùng, API của ChatGPT được sử dụng để tạo các câu trả lời có ảo giác theo khung prompt đã xây dựng, từ đó hình thành bộ dữ liệu tiêu chuẩn để đánh giá mô hình. Quá trình đánh giá giúp phát hiện và đo lường mức độ ảo giác của các mô hình ngôn ngữ lớn khác nhau, so sánh hiệu suất và xác định mô hình phù hợp nhất trong ngữ cảnh dịch vụ công.

2.5 Kết quả dự kiến

Qua những động lực nghiên cứu và phương pháp đã đặt ra, kết quả mà nhóm em mong muốn đạt được là:

- Một quy trình phù hợp nhất cho tác vụ đánh giá ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công.
- Một bộ dữ liệu tiếng Việt làm tiêu chuẩn tương ứng.
- Một bảng thống kê so sánh hiệu suất của các mô hình ngôn ngữ lớn, dựa trên bộ dữ liệu đã đưa ra.

2.6 Kế hoạch thực hiện

Giai đoạn	Công việc	Người thực hiện
01/01/2025 - 20/01/2024	Tìm hiểu, nghiên cứu các công trình liên quan đến ảo giác của mô hình ngôn ngữ lớn	Nhật, Bảo
20/01/2025 - 10/02/2025	Tìm hiểu nghiên cứu các công trình liên quan đến ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công	Nhật, Bảo
10/02/2025 - 20/02/2025	Thu thập dữ liệu chính thống về khía cạnh dịch vụ công	Nhật, Bảo
20/02/2025 - 01/03/2025	Tiền xử lý và tạo khung prompt dựa theo các công trình liên quan	Nhật, Bảo
01/03/2025 - 15/03/2025	Tạo ra các câu trả lời ảo giác tương ứng với khung prompt và bộ dữ liệu	Nhật, Bảo
15/03/2025 - 01/04/2025	Tiến hành đánh giá bộ dữ liệu thu được và tiền xử lý bổ sung	Nhật, Bảo
01/04/2025 - 01/05/2025	Áp dụng cho các mô hình ngôn ngữ lớn trong việc phát hiện ra ảo giác và thống kê so sánh	Nhật, Bảo
01/05/2025 - 01/06/2025	Tiến hành tối ưu bộ dữ liệu tiêu chuẩn, khiến cho mô hình ngôn ngữ lớn càng khó nhận diện ảo giác hơn trong ngữ cảnh dịch vụ công	Nhật, Bảo
01/06/2025 - 01/07/2025	Viết cuốn luận khóa luận tốt nghiệp và báo cáo	Nhật, Bảo

Bảng 1: Kế hoạch thực hiện

Tài liệu

[1] N. Dziri, H. Rashkin, T. Linzen, and D. Reitter, "Evaluating attribution in dialogue systems: The BEGIN benchmark," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1066–1083, 2022.

[2] H. Rashkin, V. Nikolaev, M. Lamm, M. Collins, D. Das, S. Petrov, G. S. Tomar, I. Turc, and D. Reitter, "Measuring attribution in natural language generation models," *CoRR*, vol. abs/2112.12870, 2021.

[3] N. Miao, Y. W. Teh, and T. Rainforth, "Selfcheck: Using llms to zero-shot check their own step-by-step reasoning," *ArXiv preprint*, vol. abs/2308.00436, 2023.

[4] S. Chen, Y. Zhao, J. Zhang, I.-C. Chern, S. Gao, P. Liu, and J. He, "Felm: Benchmarking factuality evaluation of large language models," *ArXiv preprint*, vol. abs/2310.00741, 2023.

[5] J. Li, X. Cheng, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Halueval: A large-scale hallucination evaluation benchmark for large language models," *CoRR*, vol. abs/2305.11747, 2023.

[6] DVC AI, "Dvc ai: Hỗ trợ dịch vụ hành chính công."

XÁC NHẬN
CỦA NGƯỜI HƯỚNG DẪN
(Ký và ghi rõ họ tên)

TP. Hồ Chí Minh, ngày 28 tháng 3 năm 2025
NHÓM SINH VIÊN THỰC HIỆN
(Ký và ghi rõ họ tên)

Mục lục

Danh sách hình

Danh sách bảng

Chương 1

Giới thiệu

1.1 Lý do chọn đề tài

Sự phát triển vượt bậc của trí tuệ nhân tạo (AI), đặc biệt trong lĩnh vực xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP), đã mở ra nhiều cơ hội ứng dụng thực tiễn. Mô hình ngôn ngữ lớn (Large Language Models - LLMs) như ChatGPT, Claude, Gemini,... đang ngày càng phổ biến trong giáo dục, y tế, truyền thông và đặc biệt là hành chính công. Với khả năng sinh ngôn ngữ tự nhiên, các mô hình này có thể trả lời câu hỏi, hướng dẫn quy trình, thậm chí hỗ trợ thực hiện thủ tục hành chính cho người dân.

Tuy nhiên, một trong những rào cản lớn nhất khiến LLMs chưa thể ứng dụng rộng rãi trong dịch vụ công là hiện tượng ảo giác thông tin (hallucination) — khi mô hình sinh ra thông tin nghe có vẻ hợp lý nhưng thực chất lại sai, bịa đặt hoặc không có căn cứ. Trong bối cảnh hành chính công, nơi thông tin cần chính xác tuyệt đối và có khả năng kiểm chứng, hiện tượng này có thể gây hậu quả nghiêm trọng, ảnh hưởng đến niềm tin người dân và tính minh bạch của cơ quan quản lý.

Mặc dù có nhiều nghiên cứu về ảo giác trong LLMs, phần lớn tập trung vào lĩnh vực y tế, khoa học hoặc kiến thức tổng quát (như Wikipedia). Hiện vẫn thiếu một bộ dữ liệu chuyên biệt phục vụ bài toán phát hiện ảo giác trong ngữ cảnh dịch vụ công tại Việt Nam. Đây chính là vấn đề còn bỏ ngỏ mà đề tài hướng tới giải quyết.

1.2 Mục tiêu và đối tượng nghiên cứu

Mục tiêu của đề tài là xây dựng một bộ dữ liệu tiếng Việt chất lượng cao, phục vụ đánh giá và phát hiện ảo giác thông tin trong phản hồi của

LLMs trong bối cảnh dịch vụ công. Đồng thời, đề tài đề xuất quy trình xây dựng dữ liệu hoàn chỉnh và thực nghiệm đánh giá nhiều LLM hiện đại để so sánh hiệu quả.

Đối tượng nghiên cứu: các mô hình ngôn ngữ lớn (GPT-4o-mini, Gemini, Claude, LLaMA,...) và các kỹ thuật sinh/đánh giá phản hồi ảo giác, với dữ liệu là các cặp hỏi–đáp phổ biến trong hệ thống hành chính công tại Việt Nam.

Phạm vi nghiên cứu: xây dựng dữ liệu tiếng Việt từ Cổng Dịch vụ công Quốc gia, thiết kế và sinh phản hồi ảo giác bằng LLMs, đánh giá khả năng phát hiện ảo giác của nhiều mô hình, không đi sâu vào fine-tuning, nhưng mở ra hướng phát triển này cho tương lai.

1.3 Mô tả bài toán

Bài toán đặt ra: Cho một câu hỏi liên quan đến dịch vụ công và một câu trả lời từ mô hình LLM, hãy xác định liệu câu trả lời này có chứa thông tin ảo giác hay không.

Yêu cầu: để giải quyết bài toán này, cần xây dựng một bộ câu hỏi – câu trả lời đúng làm chuẩn, đồng thời tạo ra một tập câu trả lời có chứa ảo giác (sinh có kiểm soát), từ đó phát triển một hệ thống đánh giá mô hình dựa trên đầu vào và phản hồi của mô hình.

Tính hấp dẫn: bài toán mang tính hấp dẫn vì hiện tại rất hiếm có nghiên cứu trong lĩnh vực xử lý ngôn ngữ tự nhiên (NLP) tập trung vào hành chính công tại Việt Nam. Nó gắn liền với nhu cầu thực tiễn về việc xây dựng các trợ lý ảo hành chính chính xác, đồng thời đặt ra thách thức lớn trong việc kiểm định tri thức pháp lý – vốn đòi hỏi độ chính xác cao và sự cập nhật liên tục.

Tính cấp thiết: một phản hồi sai từ mô hình có thể gây hiểu lầm về pháp lý hoặc khiến người dân thực hiện sai quy trình hành chính. Trong khi đó, các mô hình LLM hiện nay vẫn chưa được kiểm soát tốt trong các ngữ cảnh chuyên ngành bằng tiếng Việt.

1.4 Thách thức và hướng tiếp cận

Khó khăn:

- Thiếu benchmark cho dịch vụ công tiếng Việt.
- Phản hồi của LLM rất tự nhiên, khó phát hiện thật – giả.
- Thiếu tri thức nền thì khó xác định phản hồi có đúng không.

Các hướng tiếp cận hiện tại:

- Tạo benchmark hallucination (BEGIN, HaluEval, AIS,...).
- Gán nhãn thủ công hoặc bán tự động.
- Dùng kỹ thuật prompt engineering để sinh phản hồi sai.

Hạn chế:

- Không hỗ trợ tiếng Việt.
- Không đặt trong bối cảnh dịch vụ công.
- Thiếu phân loại pattern lỗi.
- Thiếu quy trình đánh giá có thể tái lập.

1.5 Câu hỏi nghiên cứu

- Liệu có thể xây dựng một bộ dữ liệu ảo giác tiếng Việt chất lượng cao cho dịch vụ công?
- Với prompt và pattern rõ ràng, liệu LLM có thể sinh phản hồi ảo giác hợp lý và có thể kiểm soát?
- Các mô hình hiện tại có phát hiện được ảo giác không? Có sự khác biệt nào giữa mô hình mã nguồn mở và mã nguồn đóng?
- Truyền tri thức nền có giúp mô hình cải thiện khả năng phát hiện lỗi không?

1.6 Đóng góp của đề tài

1. Xây dựng **VietPS-Hallu**: bộ dữ liệu đầu tiên về ảo giác LLM trong ngữ cảnh dịch vụ công tiếng Việt với 3.717 mẫu sinh và gán nhãn. Bộ dữ liệu này không chỉ làm nền tảng cho việc đánh giá hallucination mà còn có thể được mở rộng, cập nhật hoặc sử dụng để huấn luyện mô hình trong tương lai.

2. Đề xuất quy trình xây dựng dữ liệu rõ ràng, có thể mở rộng và tái sử dụng.

3. Thiết kế prompt và phân loại pattern lỗi phục vụ sinh và đánh giá phản hồi.Điều này góp phần làm rõ bản chất hallucination trong DVC, từ đó giúp đánh giá và phân tích lỗi có hệ thống hơn, thay vì đánh giá một cách mơ hồ.

4. Thực nghiệm nhiều LLM hiện đại, phân tích theo pattern, bộ/ngành, tri thức nền. Phân tích này giúp hiểu rõ hơn mô hình nào phù hợp trong ngữ cảnh nào, từ đó định hướng triển khai thực tế các chatbot DVC an toàn và hiệu quả hơn

Chương 2

Các công trình liên quan

Trong chương này, chúng tôi trình bày các khái niệm và công trình nghiên cứu liên quan đến đề tài khóa luận một cách khái quát nhất. Sau đó chúng tôi phân tích sâu sắc hơn các khía cạnh của khóa luận.

2.1 Hiện tượng ảo giác trong mô hình ngôn ngữ lớn

Mô hình ngôn ngữ lớn (LLMs) là các mô hình học sâu được huấn luyện trên khối lượng dữ liệu văn bản rất lớn [8], với mục tiêu chính là mô hình hóa xác suất phân phối của chuỗi từ trong ngôn ngữ tự nhiên. Các mô hình này có khả năng hiểu ngữ cảnh, tạo sinh văn bản, và thực hiện nhiều tác vụ xử lý ngôn ngữ tự nhiên (Natural Language Processing – NLP) như trả lời câu hỏi, tóm tắt văn bản, dịch máy, sinh mã lập trình, và thậm chí là lý luận theo ngữ cảnh.

Với khả năng tạo sinh và hiểu ngôn ngữ một cách linh hoạt, LLMs đang được ứng dụng rộng rãi trong các lĩnh vực như giáo dục, y tế, thương mại và đặc biệt là hành chính công. Chúng có thể hỗ trợ người dùng qua các chức năng như trả lời câu hỏi, tóm tắt văn bản, dịch máy, viết mã lập trình hoặc lý luận theo ngữ cảnh, giúp tự động hóa nhiều quy trình một cách hiệu quả.

Dù đạt được nhiều thành tựu đáng kể, LLMs vẫn tồn tại những rủi ro, đặc biệt là hiện tượng ảo giác – khi mô hình tạo ra thông tin không chính xác, sai lệch hoặc không có cơ sở thực tế [13]. Vấn đề này trở nên nghiêm trọng trong bối cảnh dịch vụ công, nơi độ chính xác và độ tin cậy của thông tin là yếu tố then chốt. Do đó, việc nghiên cứu cơ chế hoạt động và đánh giá đầu ra của LLMs là bước cần thiết nhằm đảm bảo ứng dụng an toàn và hiệu quả trong thực tiễn.

2.2 Các mô hình ngôn ngữ lớn hiện nay

Hiện nay, có rất nhiều những mô hình ngôn ngữ lớn được cho ra mắt trên toàn thế giới, trải dài từ các mô hình mã nguồn mở cho đến những mô hình mã nguồn đóng mạnh mẽ và rất phổ biến.

Các mô hình mã nguồn mở

- **Llama-3 (Meta AI)** [5]
  Mô hình ngôn ngữ thế hệ mới nhất từ Meta, thuộc dòng Llama-3, huấn luyện với tập dữ liệu chất lượng cao có bao gồm dữ liệu đa ngữ. Được thiết kế cho các tác vụ reasoning và instruction-following. Phiên bản sử dụng trong thực nghiệm là bản `Q4_K_M`, cân bằng tốt giữa dung lượng và độ chính xác, cho phép phản hồi bằng tiếng Việt khi được cung cấp prompt rõ ràng.

- **Mistral-v0.3 (Mistral AI)** [6]
  Mô hình nhỏ gọn, nổi bật với khả năng xử lý mệnh lệnh (instruct) ngắn và chính xác. Sử dụng kiến trúc decoder-only tương tự GPT, mô hình này được tối ưu hóa để hoạt động trên thiết bị cá nhân. Mistral thường phản hồi rất nhanh, phù hợp với môi trường đánh giá hàng loạt.

- **Vicuna-v1.5 (LMSys)** [15]
  Phiên bản tinh chỉnh từ LLaMA 1/2 với dữ liệu đối thoại từ ShareGPT. Vicuna 1.5 được cộng đồng đánh giá cao về khả năng phản hồi giống con người và nhất quán với truy vấn dạng hội thoại. Mô hình này thường nhạy cảm với văn phong và cấu trúc ngữ nghĩa, do đó thích hợp cho đánh giá nhị phân khi truy vấn ngắn gọn.

- **Qwen-2.5 (Alibaba Cloud)** [10]
  Dòng mô hình huấn luyện từ dữ liệu nội bộ và nguồn mở, hỗ trợ đa ngôn ngữ với trọng tâm là tiếng Trung và tiếng Anh. Tuy nhiên, trong thực nghiệm, mô hình này vẫn thể hiện năng lực phản hồi tiếng Việt đáng chú ý khi kết hợp với prompt ràng buộc chặt chẽ. Được thiết kế để phản hồi chuẩn xác trong ngữ cảnh nhúng.

- **WizardLM-2 (Microsoft Research)** [19]
  Huấn luyện với kỹ thuật *evolutionary instruction tuning*, WizardLM-2 được tối ưu hóa cho các truy vấn có logic phân tích hoặc suy luận đơn giản. Mô hình thể hiện độ nhạy cao với ngữ nghĩa câu hỏi, có thể phân biệt được mức độ lệch thông tin trong câu trả lời.

Ngoài ra, các mô hình mã nguồn mở còn được tinh chỉnh trên bộ dữ liệu tiếng Việt, cho thấy tiềm năng tốt trên một số những tác vụ và lĩnh vực nhất định.

- **Vistral (UoNLP)** [3]
  Là mô hình pretrain và fine-tune hoàn toàn trên dữ liệu tiếng Việt, tập trung vào các tác vụ đối thoại, QA, và truy vấn hành chính công. Được thiết kế nhằm cải thiện độ trôi chảy và tính chính xác khi phản hồi truy vấn tiếng Việt. Vistral là một trong số rất ít mô hình open-source huấn luyện hoàn toàn bằng tiếng Việt.

- **Qwen-Viet** [2]
  Phiên bản tiếng Việt hóa từ mô hình Qwen2.5, được tinh chỉnh với tập dữ liệu song ngữ Việt–Anh. Việc fine-tune giúp mô hình nâng cao khả năng hiểu cấu trúc văn bản hành chính Việt Nam, từ đó phản hồi chính xác hơn trong truy vấn yêu cầu nhận diện thông tin ảo giác chuyên ngành.

Các mô hình mã nguồn đóng/truy cập qua API

- **GPT-4o-mini (OpenAI)** [16]
  Phiên bản rút gọn của GPT-4o, tập trung vào tốc độ phản hồi và hiệu quả chi phí. GPT-4o-mini thừa hưởng khả năng suy luận mạnh từ GPT-4, nhưng có dung lượng nhỏ hơn, phù hợp cho các tác vụ đánh giá lặp lại như bài toán phân loại ảo giác. Mô hình hỗ trợ tiếng Việt rất tốt.

- **DeepSeek-v3-0324 (DeepSeek AI)** [4]
  Mô hình mới phát hành năm 2024, nổi bật với khả năng tổng hợp

thông tin và phản hồi chính xác trong các truy vấn logic. DeepSeek-v3 được đào tạo với dữ liệu chất lượng cao và có khả năng xử lý song ngữ. Trong các thử nghiệm nội bộ, mô hình này thể hiện sự ổn định trong đánh giá ảo giác dạng yes/no.

- **Gemini-2.0-flash (Google DeepMind)** [11]
  Phiên bản "flash" được tối ưu cho thời gian phản hồi cực nhanh, có khả năng hỗ trợ đa ngôn ngữ và tương thích tốt với các truy vấn dạng instruction. Mặc dù không chuyên sâu về tiếng Việt, nhưng Gemini vẫn có thể đưa ra phản hồi phù hợp nếu được hướng dẫn rõ ràng thông qua prompt.

- **Claude-3.5-Haiku (Anthropic)** [7]
  Một trong những mô hình có chất lượng sinh văn bản cao nhất hiện tại. Claude-3.5-Haiku hỗ trợ phân tích và đánh giá nội dung chi tiết, tuy nhiên có xu hướng phản hồi bảo thủ trong truy vấn nhị phân. Mô hình hỗ trợ tiếng Việt hạn chế, nhưng vẫn phản hồi được với ngữ cảnh hành chính công.

2.3 Các bộ dữ liệu đánh giá hallucination trong LLM

Để giải quyết hiện tượng ảo giác trong mô hình ngôn ngữ lớn, nhiều công trình nghiên cứu trước đây đã từng đưa ra các phương pháp đánh giá hiện tượng ảo giác của các mô hình ngôn ngữ lớn thông qua việc xây dựng các bộ dữ liệu tiêu chuẩn (benchmark dataset). Chẳng hạn, BEGIN là một bộ dữ liệu phân loại các câu trả lời của hệ thống đối thoại thành 3 mức độ: fully attributable (hoàn toàn được hỗ trợ bởi tri thức), not fully attributable (tạm dịch là không hoàn toàn được hỗ trợ bởi tri thức) và generic (câu trả lời chung chung) [12]. Tương tự AIS, một bộ dữ liệu được phát triển nhằm xác định liệu các tài liệu nguồn có thực sự hỗ trợ đầu ra của các mô hình tạo sinh văn bản hay không [18]. Một số bộ dữ liệu khác tập trung vào đánh giá tính thực tế ở cấp độ câu, ví dụ như SelfCheckGPT-Wikibio - một bộ dữ liệu được tạo ra bằng cách tổng hợp các bài viết từ Wikipedia với GPT-3, được chú thích thủ công về tính thực tế, gây ra thách thức cho việc phát hiện ảo giác về tiểu sử cá nhân [17].

Trong khi đó, FELM - một bộ dữ liệu được chú thích về tính thực tế trên nhiều lĩnh vực khác nhau bao gồm kiến thức thế giới, khoa học và toán học [9].

2.4 Bộ dữ liệu HaluEval

Trong khi, hầu hết các bộ dữ liệu trên dựa vào tri thức tham chiếu rõ ràng hoặc nguồn dữ liệu có cấu trúc, thì HaluEval là một tập benchmark quy mô lớn được xây dựng nhằm đánh giá khả năng nhận diện ảo giác trong phản hồi sinh ra bởi các mô hình ngôn ngữ lớn bằng chính mô hình ngôn ngữ lớn [14]. Nhóm tác giả tập trung vào việc xác định loại nội dung nào và mức độ nào khiến các mô hình như ChatGPT dễ sinh ra thông tin sai lệch hoặc không thể kiểm chứng.

Để tạo dữ liệu huấn luyện, nhóm nghiên cứu đề xuất quy trình hai bước: lấy mẫu và lọc mẫu. Trong đó, các phản hồi từ LLM được tạo ra từ nhiều chỉ dẫn khác nhau, sau đó được lọc để chọn ra những phản hồi khó đánh giá và dễ sinh hallucination. Bên cạnh quy trình tạo mẫu tự động, nhóm tác giả cũng mời người gán nhãn chuyên trách để gán nhãn thủ công các phản hồi của ChatGPT, xác định chính xác các nội dung bị ảo giác.

Kết quả thực nghiệm cho thấy khoảng 19,5% phản hồi của ChatGPT chứa nội dung không thể kiểm chứng, thường liên quan đến các chủ đề đặc thù. Đồng thời, các LLM hiện tại gặp nhiều khó khăn trong việc phát hiện các ảo giác ngôn ngữ, dù một số chiến lược như bổ sung tri thức bên ngoài hoặc suy luận theo bước (reasoning) có thể cải thiện hiệu quả phát hiện.

HaluEval vì vậy là một nguồn dữ liệu quan trọng để hiểu rõ hơn về các dạng hallucination, cũng như đánh giá và phát triển các mô hình ngôn ngữ lớn đáng tin cậy và an toàn hơn trong tương lai.

2.5 Chatbot Dịch vụ công tiếng Việt

Hiện nay, một số hệ thống ứng dụng mô hình ngôn ngữ lớn (LLM) vào lĩnh vực dịch vụ công đã được triển khai trong thực tế, tiêu biểu là chatbot Dichvucong.me [1]. Hệ thống này hỗ trợ người dùng tra cứu thông tin về các thủ tục hành chính phổ biến như cấp Căn cước công dân, làm hộ

chiếu, đăng ký khai sinh... với giao diện đơn giản và phản hồi bằng tiếng Việt.

Tuy nhiên, hiện chưa có công bố học thuật nào đánh giá khả năng của các mô hình đứng sau chatbot này trong việc phát hiện hoặc hạn chế hiện tượng ảo giác (hallucination). Điều này cho thấy một khoảng trống quan trọng: dù LLM có thể sinh phản hồi mạch lạc, nhưng không rõ liệu chúng có thể tự đánh giá và nhận diện được các lỗi hallucination của chính mình hay không. Đề tài này hướng đến giải quyết khoảng trống đó thông qua việc xây dựng bộ dữ liệu có kiểm soát nhằm đo lường khả năng nhận diện hallucination của mô hình, thay vì chỉ đánh giá đầu ra như các hướng tiếp cận truyền thống.

Chương 3

Xây dựng bộ dữ liệu

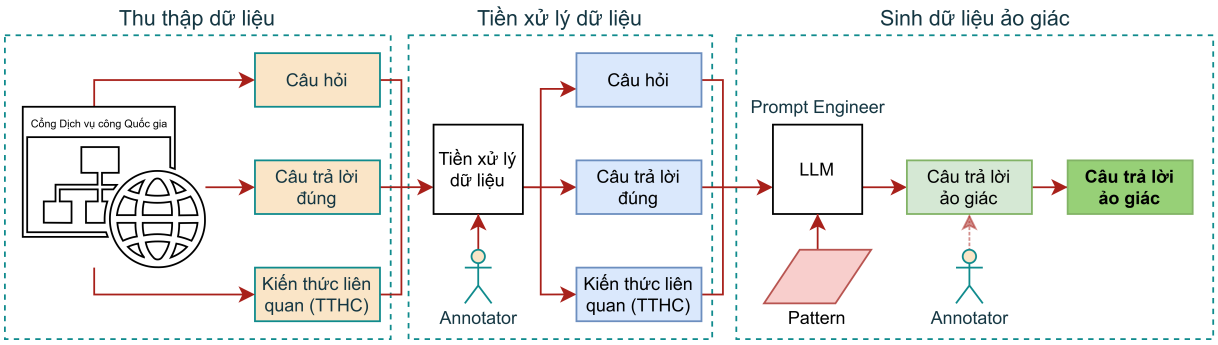
3.1 Tổng quan quy trình xây dựng bộ dữ liệu

Việc đánh giá hiện tượng ảo giác trong các mô hình ngôn ngữ lớn (LLMs) đòi hỏi một bộ dữ liệu được thiết kế bài bản và chính xác. Trong ngữ cảnh dịch vụ công, bộ dữ liệu còn đòi hỏi cần phải phản ánh được thực tế các vấn đề của công dân, có thể truy xuất được nguồn gốc của những thông tin liên quan và mức độ phù hợp về mặt pháp lý. Để giải quyết vấn đề trên, chúng tôi đề xuất một quy trình xây dựng bộ dữ liệu ảo giác có giám sát, hỗ trợ tiếng Việt. Quy trình sẽ gồm có 3 công đoạn chính, bao gồm: (i) Tìm kiếm và thu thập dữ liệu; (ii) Tiền xử lý dữ liệu cùng với kiểm tra chính tả bằng annotator; (iii) Sinh dữ liệu ảo giác kết hợp với chú thích thủ công bằng annotator. Sơ đồ tổng quát của quy trình được thể hiện ở Hình 3.1, thiết kế quy trình theo từng bước rõ ràng giúp đảm bảo tính tự động hóa, khả năng tái lập, và đặc biệt là có thể đánh giá chính xác các mô hình ngôn ngữ lớn trong ngữ cảnh có yêu cầu đặc thù cao như dịch vụ công, nổi trội với khả năng hỗ trợ tiếng Việt. Các thông tin mô tả chi tiết về bộ dữ liệu (giải thích các trường dữ liệu) được trình bày ở phần phụ lục A. Cụ thể, các bước trong quy trình bao gồm:

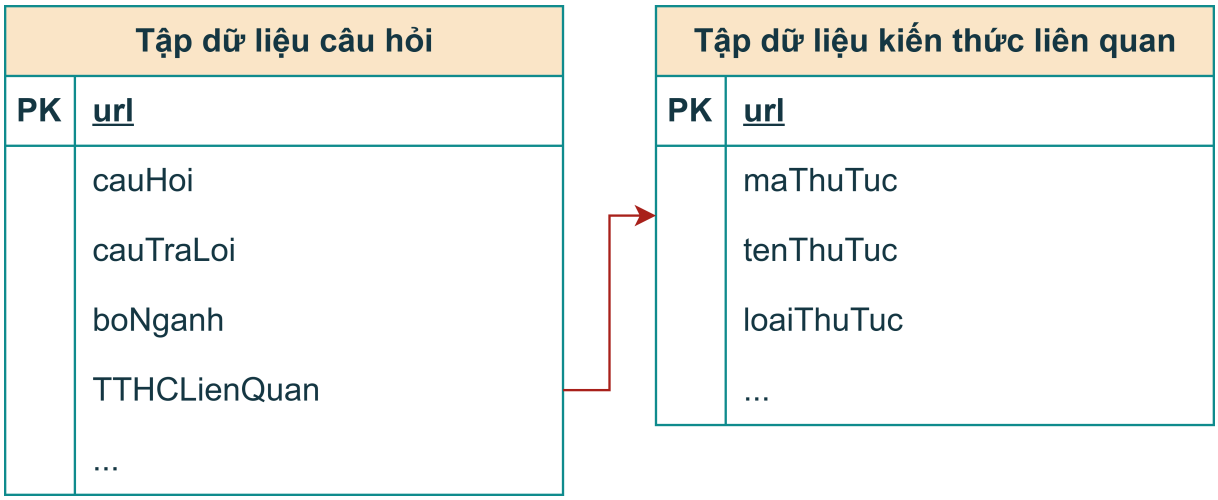
Bước 1: Tìm kiếm và thu thập dữ liệu

Để phục vụ cho bài toán phát hiện ảo giác trong mô hình ngôn ngữ lớn, đề tài lựa chọn **Cổng Dịch vụ công Quốc gia** làm nguồn dữ liệu chính thống. Đây là hệ thống do Văn phòng Chính phủ quản lý, cung cấp thông tin cập nhật về các thủ tục hành chính (TTHC) trên cả nước. Từ cổng này, hai tập dữ liệu chính đã được thu thập:

- Tập dữ liệu câu hỏi: gồm các câu hỏi thường gặp, đi kèm câu trả lời



Hình 3.1: Quy trình xây dựng bộ dữ liệu ảo giác trong ngữ cảnh dịch vụ công



Hình 3.2: Lược đồ quan hệ các tập dữ liệu có trong quy trình

chính thức và danh sách các TTHC có liên quan.

- Tập dữ liệu kiến thức liên quan (TTHC): gồm các thủ tục hành chính, mỗi thủ tục chứa đầy đủ thông tin như điều kiện, hồ sơ, cơ sở pháp lý, v.v.

Mỗi câu hỏi có thể tham chiếu đến một hoặc nhiều thủ tục hành chính liên quan. Hình 3.2 thể hiện mối quan hệ giữa các tập dữ liệu trong bộ dữ liệu sau khi thu thập. Các TTHC này được xem như tri thức nền cho câu hỏi và sẽ đóng vai trò quan trọng trong quá trình sinh và đánh giá phản hồi ảo giác ở các bước tiếp theo.

Bước 2: Tiền xử lý dữ liệu

Trước khi tiến hành sinh dữ liệu ảo giác phục vụ đánh giá mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công, bước tiền xử lý dữ liệu có vai trò quan trọng nhằm chuẩn hóa và làm sạch tập dữ liệu đầu vào. Các câu hỏi và câu trả lời thu thập từ Cổng Dịch vụ công Quốc gia tồn tại nhiều lỗi như: thiếu thông tin, trùng lặp nội dung, sai chính tả, hoặc thiếu tính thống nhất. Do đó, từ tập dữ liệu thô ban đầu, quá trình tiền xử lý bao gồm:

- Lọc trùng lặp bằng cách tính độ tương đồng cosine giữa các cặp câu hỏi và câu trả lời, loại bỏ bản ghi có độ tương đồng lớn hơn một ngưỡng nhất định.
- Ưu tiên giữ lại các câu có đầy đủ thông tin về bộ/ngành và nhiều thủ tục hành chính liên quan.
- Loại bỏ bản ghi thiếu thông tin: chỉ giữ lại các câu hỏi có đủ ba thành phần: câu trả lời đúng, bộ/ngành, và tri thức liên quan.
- Giữ tạm bản ghi thiếu bộ/ngành để dự phòng trong trường hợp cần bổ sung dữ liệu theo lĩnh vực.

Bước 3: Kiểm tra dữ liệu thủ công

Sau khi hoàn tất tiền xử lý tự động, đề tài tiếp tục thực hiện bước kiểm tra dữ liệu thủ công nhằm tăng tính chính xác cho bộ dữ liệu. Hai annotator độc lập được phân công với nhiệm vụ chính như sau:

- Kiểm tra chính tả của toàn bộ câu hỏi và câu trả lời.
- Phát hiện lỗi logic hoặc lỗi ngữ nghĩa còn tồn tại trong dữ liệu đã lọc.
- Đảm bảo mỗi bản ghi trong bộ dữ liệu câu hỏi - câu trả lời và bộ dữ liệu kiến thức liên quan có định dạng phù hợp.

Các sửa đổi được thống nhất và hợp nhất thành bộ dữ liệu đầu vào sạch cuối cùng, làm cơ sở cho bước sinh phản hồi ảo giác tiếp theo.

Bước 4: Sinh dữ liệu ảo giác

Sau khi hoàn tất bước tiền xử lý, dữ liệu được đưa vào quá trình sinh câu trả lời ảo giác – bước quan trọng nhất trong toàn bộ quy trình xây dựng bộ dữ liệu. Mục tiêu là tạo ra các phản hồi sai lệch nhưng có vẻ hợp lý, nhằm phục vụ cho đánh giá và huấn luyện các mô hình phát hiện ảo giác.

Song song với việc lựa chọn một mô hình ngôn ngữ lớn để thực hiện sinh dữ liệu ảo giác, ta cần phải lựa chọn những siêu tham số và mẫu truy vấn phù hợp. Các siêu tham số chính gồm: `temperature`, `top_p`, và `max_output_tokens`, nhằm khuyến khích tính ngẫu nhiên và đa dạng trong phản hồi, đồng thời duy trì đảm bảo độ dài đầu ra tương xứng với phản hồi gốc.

Để kiểm soát chất lượng và hướng dẫn mô hình sinh đúng mục tiêu, đề tài thiết kế một hệ thống truy vấn (prompt) rõ ràng gồm các thành phần: mô tả mục tiêu (instruction), phân loại lỗi (pattern), và dữ liệu đầu vào. Bốn pattern ảo giác được áp dụng ngẫu nhiên trên từng câu hỏi bao gồm: (i) hiểu sai ngữ cảnh; (ii) mâu thuẫn với tri thức; (iii) quá chung hoặc quá chi tiết; (iv) suy luận sai từ tri thức.

Bước 5: Chú thích dữ liệu ảo giác

Sau khi hoàn tất quá trình sinh phản hồi ảo giác, đề tài thực hiện bước gán nhãn thủ công toàn bộ tập phản hồi nhằm đảm bảo tính chính xác và độ tin cậy. Đây là bước xác nhận rằng liệu các câu trả lời đã sinh ra có thực sự là phản hồi ảo giác hay không.

Toàn bộ các phản hồi ảo giác được đánh giá bởi nhóm nghiên cứu thông qua việc đọc nội dung câu hỏi, câu trả lời đúng, tri thức liên quan và phản hồi sinh ra. Dựa trên hiểu biết pháp lý và ngữ cảnh dịch vụ công, bất kỳ sai sót nào về số liệu, ngữ nghĩa hay lập luận đều được xem là biểu hiện ảo giác.

Việc chú thích này được thực hiện thủ công bởi những người có kiến thức về miền dịch vụ công, đảm bảo tính nhất quán và chất lượng của bộ dữ liệu. Kết quả là một tập phản hồi ảo giác đã qua xác thực, phục vụ trực tiếp cho các bước đánh giá mô hình và xây dựng hệ thống phát hiện ảo giác sau này.

3.2 Tìm kiếm và thu thập dữ liệu

Quá trình thu thập dữ liệu là bước đầu tiên và cũng là nền tảng quan trọng của toàn bộ quy trình xây dựng bộ dữ liệu phát hiện ảo giác trong ngữ cảnh dịch vụ công. Nhằm đảm bảo độ tin cậy, chính xác và đầy đủ của dữ liệu, đề tài đã xây dựng một quy trình thu thập có giám sát, được triển khai thông qua một chuỗi các notebook Python sử dụng thư viện `selenium`, `pandas`, `BeautifulSoup` và các công cụ hỗ trợ xử lý dữ liệu khác. Tổng cộng, giai đoạn này bao gồm 7 notebook chính, tương ứng với 7 công đoạn theo thứ tự dưới đây:

3.2.1 Thiết lập môi trường

Trước khi tiến hành thu thập và xử lý dữ liệu, chúng tôi đã tiến hành thiết lập một môi trường làm việc ổn định và tái lập. Việc chuẩn bị kỹ lưỡng từ đầu đảm bảo tính thống nhất, khả năng mở rộng và hạn chế rủi ro trong quá trình thực thi các tác vụ tự động.

a) Môi trường và công nghệ sử dụng

Tất cả các bước thu thập và xử lý dữ liệu đều được thực hiện trên nền tảng Python 3, sử dụng môi trường tương thích với Jupyter Notebook. Lựa chọn Python giúp tận dụng sức mạnh từ các thư viện mã nguồn mở hiện đại, phục vụ cho việc tự động hóa trình duyệt, xử lý dữ liệu bảng và tối ưu hiệu năng.

b) Các thư viện và công cụ chính

Để đảm bảo hiệu quả cho từng giai đoạn trong pipeline, các nhóm thư viện chính được sử dụng bao gồm:

- Tự động hóa và thu thập dữ liệu web: `selenium`, `webdriver_manager` – hỗ trợ điều khiển trình duyệt web tự động, phù hợp với các trang web có nội dung động.
- Xử lý dữ liệu bảng: `pandas` – thao tác với dữ liệu CSV, lọc, gộp và xử lý tiền xử lý nhanh chóng.
- Hiển thị và theo dõi tiến độ: `tqdm` – cung cấp thanh tiến trình giúp quan sát thời gian chạy các vòng lặp lớn.
- Hỗ trợ định dạng dữ liệu Excel và các tệp phụ trợ: `openpyxl`, `IPython.display`, ...

Việc cài đặt và cấu hình các thư viện được thực hiện tập trung trong một notebook khởi tạo (`setup.ipynb`), đảm bảo khi chạy lại các bước sau sẽ không gặp lỗi thiếu thư viện.

c) Cấu hình trình duyệt và hệ thống

Trình duyệt web được sử dụng trong việc thu thập dữ liệu là Google Chrome, điều khiển thông qua ChromeDriver với chế độ không giao diện (headless) để tối ưu tốc độ xử lý và hạn chế tương tác thủ công. Việc này cho phép các tác vụ tự động có thể thực hiện được cả trong môi trường máy chủ hoặc nền tảng cloud.

ChromeDriver được cài đặt và đồng bộ hoàn toàn tự động thông qua `webdriver_manager`, giúp loại bỏ sai lệch giữa phiên bản trình duyệt và trình điều khiển.

d) Tổ chức thư mục

Cấu trúc thư mục được chuẩn hóa từ đầu để đảm bảo tổ chức khoa học và dễ dàng truy xuất:

- `raw_data/`: Chứa toàn bộ dữ liệu gốc được thu thập (CSV, JSON).
- `backup/`: Lưu trữ bản sao trung gian nhằm dự phòng trong trường hợp mất dữ liệu.
- `old_csv_test/`: Thư mục phụ trợ trong quá trình thử nghiệm hoặc kiểm tra dữ liệu cũ.
- Các notebook chính (`.ipynb`): Được chạy lần lượt theo trình tự thực thi như trong file `README.md`.

Notebook `setup.ipynb` đóng vai trò như một bước chuẩn hóa ban đầu, đảm bảo các điều kiện kỹ thuật cần thiết cho toàn bộ quy trình thu thập và xử lý dữ liệu được thực hiện hiệu quả, nhất quán và có thể tái lập.

3.2.2 Thu thập các đường dẫn câu hỏi

Sau khi thiết lập môi trường, bước tiếp theo trong quy trình là thu thập toàn bộ các đường dẫn (URL) dẫn đến các câu hỏi thường gặp (FAQ) từ **Cổng Dịch vụ công Quốc gia** tại địa chỉ:

`https://dichvucong.gov.vn/p/home/dvc-cau-hoi-pho-bien.html`

Đây là nguồn dữ liệu chính thống, chứa hàng nghìn câu hỏi – câu trả lời thuộc các lĩnh vực khác nhau như: đăng ký hộ tịch, giấy phép kinh doanh, bảo hiểm xã hội, đất đai, xây dựng, giáo dục, y tế,... được phân loại rõ ràng theo bộ/ngành và đối tượng sử dụng (Công dân, Doanh nghiệp, Tổ chức khác).

a) Mục tiêu thu thập

Mục tiêu của bước này là lấy toàn bộ đường dẫn dẫn tới các câu hỏi thường gặp (FAQ) trên Cổng Dịch vụ công Quốc gia, được phân loại theo bộ/ngành và nhóm đối tượng sử dụng (Công dân, Doanh nghiệp, Tổ chức khác). Việc thu thập đường dẫn đầy đủ và chính xác là nền tảng để có thể truy xuất nội dung câu hỏi, câu trả lời chính thức và thông tin liên quan ở các bước tiếp theo.

b) Phương pháp thu thập

Toàn bộ quá trình được thực hiện trong notebook `selenium_crawler.ipynb`, sử dụng thư viện Selenium để tự động hóa thao tác trình duyệt và tương tác với các thành phần động của giao diện. Hệ thống đầu tiên truy cập vào danh sách bộ/ngành được công bố trên trang, sau đó lần lượt đi vào từng bộ/ngành và từng phân loại người dùng để lấy toàn bộ URL của các câu hỏi hiển thị.

Khi thu thập từng câu hỏi, chương trình đồng thời ghi nhận thông tin về bộ/ngành và phân loại người dùng tương ứng. Cơ chế phân trang cũng được xử lý đầy đủ nhằm đảm bảo không bỏ sót bất kỳ URL nào.

c) Kết quả đầu ra

Kết quả thu thập được là một tập hợp các file CSV:

- Các file `{TenBoNganh}_link.csv` và `{LoaiToChuc}Tab_link.csv` chứa danh sách URL câu hỏi tương ứng với từng bộ/ngành hoặc phân loại.
- File `ministries.csv` ghi lại danh sách đầy đủ các bộ/ngành hiện có.

Dữ liệu đầu ra của bước này là danh sách toàn diện các đường dẫn đến các câu hỏi, đóng vai trò làm đầu vào cho bước crawl nội dung chi tiết ở phần tiếp theo.

3.2.3 Thu thập nội dung chi tiết của từng câu hỏi

Sau khi thu thập toàn bộ đường dẫn tới các câu hỏi từ Cổng Dịch vụ công Quốc gia, bước tiếp theo là thu thập nội dung chi tiết tại từng URL.

a) Mục tiêu thu thập

Sau khi đã thu thập danh sách các đường dẫn đến từng câu hỏi, bước tiếp theo là trích xuất đầy đủ nội dung tương ứng với từng đường dẫn đó. Mỗi câu hỏi cần được thu thập kèm theo các thành phần: nội dung câu hỏi, câu trả lời chính thức, thông tin bộ/ngành, và các liên kết đến thủ tục hành chính liên quan (tri thức nền).

b) Phương pháp thu thập

Bước này được triển khai trong notebook `link_detail_crawler.ipynb`, tiếp tục sử dụng thư viện Selenium để duyệt vào từng trang chi tiết câu hỏi. Với mỗi đường dẫn, chương trình truy cập và trích xuất các trường thông tin gồm:

- Câu hỏi (tiêu đề).
- Câu trả lời (dạng văn bản đầy đủ).
- Danh sách URL dẫn đến các TTHC liên quan.
- Danh sách URL dẫn đến các câu hỏi liên quan.

Để xử lý các trường hợp trang bị lỗi hoặc nội dung không tải được, chương trình thiết kế thêm các cơ chế timeout và retry, đồng thời ghi nhận danh sách các URL lỗi để xử lý lại về sau. Những bản ghi thiếu toàn bộ nội dung sẽ bị loại bỏ để đảm bảo độ sạch của dữ liệu.

c) Kết quả đầu ra

Quá trình crawl tạo ra hai tệp dữ liệu chính. Tệp `link_detail.csv` chứa toàn bộ thông tin chi tiết của từng câu hỏi, bao gồm nội dung, câu trả lời và bộ/ngành. Tệp `tthc_link.csv` ánh xạ giữa từng câu hỏi và các TTHC liên quan đến nó. Ngoài ra, hai bản sao dự phòng là `link_detail_copy.csv` và `raw_data/link_detail.csv` được lưu để đảm bảo an toàn dữ liệu trong quá trình xử lý sau.

Tập dữ liệu thu được ở bước này là đầu vào trực tiếp cho bước crawl chi tiết nội dung của các thủ tục hành chính ở phần tiếp theo.

3.2.4 Thu thập nội dung chi tiết của từng thủ tục hành chính (TTHC)

Các thủ tục hành chính (TTHC) đóng vai trò là tri thức liên quan cho mỗi câu hỏi – câu trả lời. Trong ngữ cảnh phát hiện ảo giác, việc cung cấp thông tin nền về TTHC giúp làm cơ sở để đánh giá một phản hồi là đúng hay mang tính bịa đặt. Do đó, bước thu thập nội dung chi tiết của từng TTHC là yếu tố không thể thiếu trong quy trình xây dựng bộ dữ liệu.

a) Mục tiêu thu thập

Mục tiêu của bước này là trích xuất đầy đủ nội dung chi tiết của từng thủ tục hành chính (TTHC) được liên kết trong các câu hỏi đã thu thập ở bước trước. Mỗi TTHC được xem là một đơn vị tri thức độc lập, đóng vai trò làm nền tảng pháp lý cho việc sinh và đánh giá phản hồi trong ngữ cảnh dịch vụ công. Việc thu thập đầy đủ các trường thông tin của TTHC giúp xây dựng kho tri thức chuẩn hóa phục vụ cho các giai đoạn sau.

b) Phương pháp thu thập

Quá trình thu thập được thực hiện trong notebook `tthc_crawler.ipynb`, sử dụng Selenium để truy cập tự động vào từng URL TTHC thu được từ tệp `tthc_link.csv` ở bước trước. Với mỗi URL, chương trình trích xuất các trường thông tin quan trọng bao gồm:

- Mã thủ tục, tên thủ tục, loại thủ tục, số quyết định, cấp thực hiện.
- Lĩnh vực, trình tự thực hiện, cách thức thực hiện, thành phần hồ sơ.
- Đối tượng thực hiện, cơ quan thực hiện, cơ quan có thẩm quyền.
- Địa chỉ tiếp nhận hồ sơ, cơ quan được ủy quyền, cơ quan phối hợp.
- Kết quả thực hiện, căn cứ pháp lý, yêu cầu và điều kiện thực hiện.
- Từ khóa và mô tả của thủ tục hành chính.

Các trường thông tin được thu thập dưới dạng văn bản thô, chưa qua xử lý chuẩn hóa, nhằm giữ lại đầy đủ thông tin gốc phục vụ cho việc tổ chức lại thành tri thức trong các bước kế tiếp.

Một số URL có thể dẫn đến trang lỗi hoặc bị thiếu thông tin do lỗi hệ thống hoặc nội dung không còn tồn tại. Những URL như vậy được ghi nhận lại để xử lý bổ sung ở bước crawl bổ sung sau này.

c) Kết quả đầu ra

Sau quá trình thu thập, hai tệp dữ liệu chính được tạo ra. Tệp `tthc_detail.csv` chứa nội dung chi tiết các TTHC dưới dạng bảng có cấu trúc. Tệp `tthc_detail.json` chứa nội dung tương tự nhưng được tổ chức dưới dạng dữ liệu JSON linh hoạt, phục vụ cho việc xử lý văn bản tự do trong các mô hình ngôn ngữ.

Hai bản sao dữ liệu dự phòng là `tthc_detail_copy.csv` và `raw_data/tthc_detail.csv` cũng được lưu lại để phục vụ cho bước tổng hợp sau cùng. Tập dữ liệu này là đầu vào quan trọng cho các bước xử lý tri thức, tổ chức lại thông tin và sinh dữ liệu ảo giác.

3.2.5 Thu thập bổ sung các thủ tục hành chính bị thiếu

Trong quá trình thu thập thông tin chi tiết các thủ tục hành chính (TTHC), một số URL gặp lỗi hoặc thiếu dữ liệu do cấu trúc trang không đồng nhất, lỗi kỹ thuật trong quá trình tải trang, hoặc một số TTHC chỉ xuất hiện ở phần liên kết sâu của câu hỏi. Để khắc phục, một bước thu thập bổ sung được thực hiện nhằm đảm bảo độ bao phủ đầy đủ cho toàn bộ tập tri thức.

Notebook `full_crawler.ipynb` được thiết kế để tự động rà soát lại tất cả các URL TTHC có liên kết trong tập câu hỏi, so sánh với tập đã crawl trước đó và phát hiện các URL chưa có dữ liệu. Những URL thiếu được tái crawl bằng Selenium, có bổ sung cơ chế log và timeout để tránh lỗi hệ thống.

Kết quả được lưu tại hai tệp: `tthc_recrawl.csv` chứa dữ liệu đã thu thập lại thành công, và bản sao dự phòng `raw_data/tthc_recrawl.csv`. Tập dữ liệu này sau đó được hợp nhất với `tthc_detail.csv` để đảm bảo

không bỏ sót bất kỳ tri thức nào.

Bước crawl bổ sung giúp đảm bảo tính toàn vẹn và độ tin cậy của dữ liệu đầu vào, đặc biệt quan trọng trong bối cảnh đánh giá hiện tượng ảo giác của các mô hình ngôn ngữ lớn, nơi việc thiếu một tri thức liên quan có thể gây sai lệch kết quả đánh giá.

3.2.6 Tổng hợp liên kết và thông tin phân loại

Sau khi hoàn tất thu thập nội dung các câu hỏi và thủ tục hành chính (TTHC) liên quan, bước tiếp theo là chuẩn hóa và gắn nhãn phân loại cho từng câu hỏi. Mục tiêu là tạo ra một bảng hợp nhất, trong đó mỗi câu hỏi được liên kết rõ ràng với bộ/ngành phụ trách, loại người dùng (Công dân, Doanh nghiệp, Tổ chức khác), và danh sách các TTHC liên quan.

Notebook `link_type_extractor.ipynb` thực hiện việc trích xuất, chuẩn hóa và gộp dữ liệu từ nhiều nguồn đầu vào như `ministries.csv`, các file URL phân loại theo bộ/ngành hoặc người dùng, và `link_detail.csv`. Quá trình xử lý bao gồm việc đồng bộ định dạng, nối dữ liệu theo khóa URL, tách cột phân loại, và loại bỏ các bản ghi lỗi hoặc trùng lặp.

Kết quả là tệp `link_type.csv`, trong đó mỗi dòng đại diện cho một câu hỏi, kèm đầy đủ metadata như tiêu đề, bộ/ngành, loại người dùng và danh sách các URL TTHC liên quan. Bản sao dự phòng được lưu tại `raw_data/link_type.csv`.

Tập dữ liệu này đóng vai trò quan trọng trong các bước tiền xử lý và tổng hợp, giúp liên kết rõ ràng giữa nội dung câu hỏi và tri thức liên quan, đồng thời hỗ trợ phân tích theo chiều ngành, người dùng hoặc kiểu lỗi sinh ra trong quá trình đánh giá mô hình.

3.2.7 Tổng hợp dữ liệu thô

Sau khi hoàn tất các bước thu thập thành phần bao gồm nội dung câu hỏi – câu trả lời, thông tin thủ tục hành chính (TTHC), metadata phân loại và các bản ghi bổ sung, bước cuối cùng trong giai đoạn thu thập là tổng hợp toàn bộ dữ liệu vào hai bảng chuẩn hóa. Notebook `raw_data_aggregator.ipynb` thực hiện việc gộp dữ liệu từ các nguồn: `link_detail.csv`, `tthc_detail.csv`, `tthc_recrawl.csv`, và `link_type.csv`. Sau khi

loại bỏ trùng lặp, chuẩn hóa thông tin và xây dựng liên kết giữa câu hỏi và danh sách TTHC, hai bảng chính được tạo ra: `raw_link.csv` chứa thông tin chi tiết từng câu hỏi và metadata liên quan, và `raw_tthc.csv` chứa nội dung tri thức (TTHC) đầy đủ. Hai bảng này đóng vai trò là đầu vào chính thức cho các bước sinh phản hồi ảo giác và đánh giá mô hình trong các phần sau, với cấu trúc rõ ràng, truy xuất dễ dàng và khả năng mở rộng linh hoạt.

Cụ thể, số lượng bản ghi ở 2 tập dữ liệu thô sau khi thu thập như sau:

- Tập dữ liệu câu hỏi (`raw_link.csv`): 9452 mẫu.
- Tập dữ liệu kiến thức liên quan (`raw_tthc.csv`): 2695 mẫu.

Các thông tin mô tả chi tiết về bộ dữ liệu (giải thích các trường dữ liệu) được trình bày ở phần phụ lục A. Lần cập nhật dữ liệu thô cuối cùng: 10/05/2025.

3.3 Tiền xử lý dữ liệu

Sau khi hoàn tất quá trình thu thập dữ liệu thô từ nhiều tập, dữ liệu đầu vào vẫn còn tồn tại nhiều vấn đề như: dữ liệu trùng lặp, thiếu thông tin, chưa đồng nhất định dạng, hoặc nội dung chưa phù hợp cho quá trình sinh phản hồi ảo giác. Vì vậy, đề tài tiến hành giai đoạn tiền xử lý nhằm chuẩn hóa và nâng cao chất lượng tập dữ liệu trước khi chuyển sang giai đoạn sinh và đánh giá.

Quá trình tiền xử lý được tổ chức theo 4 bước độc lập, thực hiện tuần tự qua các notebook: `preprocess_1.ipynb`, `preprocess_2.ipynb`, `preprocess_3.ipynb`, và `preprocess_tthc.ipynb`. Dữ liệu đầu vào chính bao gồm:

- `raw_link.csv`: chứa câu hỏi, câu trả lời, bộ/ngành, loại người dùng và danh sách URL TTHC liên quan.
- `raw_tthc.csv`: chứa toàn bộ thông tin về các thủ tục hành chính.

3.3.1 Lọc dữ liệu trùng lặp và chuẩn hóa sơ bộ

Sau khi tổng hợp dữ liệu thô, bước đầu tiên trong quy trình tiền xử lý là loại bỏ các mẫu bị trùng lặp và chuẩn hóa sơ bộ nội dung văn bản để đảm bảo tính nhất quán trước khi đi vào các bước xử lý sâu hơn.

a) Dữ liệu đầu vào

- Tệp sử dụng: `raw_link.csv`.
- Các trường thông tin cần chú ý: câu hỏi, câu trả lời đúng, danh sách thủ tục hành chính liên quan, bộ/ngành,...
- Số lượng bản ghi: 9452 mẫu.

b) Mục tiêu xử lý

- Loại bỏ các mẫu dữ liệu trùng nhau về nội dung (câu hỏi hoặc câu trả lời) dựa trên độ tương đồng cosine.
- Ưu tiên giữ lại các bản ghi có nhiều TTHC liên quan hơn trong trường hợp bị trùng.
- Làm sạch và chuẩn hóa sơ bộ định dạng văn bản ở cả câu hỏi và câu trả lời.

c) Phương pháp xử lý

1. Dữ liệu đầu vào được chuẩn hóa bằng cách loại bỏ các thẻ HTML, ký hiệu đặc biệt (`\ `, `\&quot;`, `<br>`...), dòng trống và các cụm từ dư thừa như "`Câu hỏi:`", "`Câu trả lời:`".
2. Sử dụng TF-IDF Vectorizer để mã hóa văn bản và tính ma trận cosine similarity giữa các câu hỏi (`question_text`) và các câu trả lời (`answer_text`).
3. Đặt ngưỡng cosine = 0.95 để phát hiện các cặp trùng lặp.

4. Với các cặp bị trùng, chọn giữ lại bản ghi có nhiều thủ tục hành chính liên quan hơn (dựa vào độ dài của `tthc_list`). Nếu bằng nhau, giữ bản ghi xuất hiện đầu tiên theo chỉ số dòng.

d) Kết quả đầu ra

- Tệp trả về: `preprocess_1.csv`.
- Số lượng bản ghi còn lại: 9289 mẫu.
- Dữ liệu được đảm bảo không có trùng lặp ngữ nghĩa và đã được làm sạch sơ bộ về mặt văn bản.

e) Nhận xét

Việc loại bỏ trùng lặp và chuẩn hóa định dạng văn bản giúp dữ liệu được đảm bảo không có trùng lặp ngữ nghĩa và đã được làm sạch sơ bộ về mặt văn bản từ đó khiến cho đầu vào nhất quán và chất lượng hơn cho các bước tiếp theo trong pipeline. Ngoài ra, việc giữ lại các bản ghi có nhiều TTHC liên quan góp phần tăng cường độ phủ kiến thức, phục vụ tốt hơn cho giai đoạn sinh phản hồi ảo giác và đánh giá mô hình.

3.3.2 Lọc các bản ghi thiếu thông tin

Sau khi loại bỏ dữ liệu trùng lặp và chuẩn hóa sơ bộ, bước tiếp theo trong giai đoạn tiền xử lý là lọc các bản ghi không đầy đủ thông tin — cụ thể là những mẫu thiếu câu trả lời, thiếu danh sách TTHC liên quan hoặc có lỗi định dạng nghiêm trọng. Việc đảm bảo tính đầy đủ của dữ liệu đầu vào là cần thiết để phục vụ cho các giai đoạn sinh phản hồi ảo giác và đánh giá mô hình ngôn ngữ lớn.

a) Dữ liệu đầu vào

- Tệp sử dụng: `preprocess_1.csv`.
- Các trường thông tin cần chú ý: câu hỏi, câu trả lời đúng, danh sách thủ tục hành chính liên quan, bộ/ngành,...

- Số lượng bản ghi: 9289 mẫu.

b) Mục tiêu xử lý

- Loại bỏ các bản ghi thiếu nội dung câu hỏi hoặc câu trả lời.
- Loại bỏ các mẫu không có tri thức liên quan (`tthc_list` rỗng hoặc sai định dạng).
- Đảm bảo định dạng JSON hợp lệ đối với trường `tthc_list`.
- Làm sạch các lỗi text phổ biến như nội dung trống, chứa chuỗi `"null"`, `"Không có thông tin"`,...

c) Phương pháp xử lý

1. Với mỗi bản ghi, kiểm tra độ dài của `question_text` và `answer_text`, loại bỏ nếu nhỏ hơn 10 ký tự đồng thời loại bỏ các ký tự đặc biệt.
2. Kiểm tra trường `tthc_list` xem có hợp lệ hay không bằng cách kiểm tra từng phần phần tử của danh sách (chính là mỗi thủ tục hành chính) có nằm trong danh sách thủ tục hành chính của bộ dữ liệu kiến thức liên quan ban đầu hay không.
3. Loại bỏ các bản ghi có nội dung câu hỏi bị thiếu hoặc không hợp lệ.
4. Sau khi lọc, reset lại chỉ số dòng và lưu ra file mới.

d) Kết quả đầu ra

- Tệp trả về: `preprocess_2.csv`.
- Số lượng bản ghi còn lại: 9205 mẫu.

e) Nhận xét

Tất cả các bản ghi còn lại được đảm bảo có đầy đủ nội dung câu hỏi, câu trả lời và danh sách thủ tục hành chính hợp lệ. Đây là bước lọc quan

trọng để đảm bảo tính toàn vẹn và độ tin cậy cho dữ liệu huấn luyện và đánh giá. Việc loại bỏ các bản ghi thiếu thông tin hoặc không hợp lệ giúp tăng độ chính xác trong quá trình sinh phản hồi ảo giác và đảm bảo kết quả đánh giá không bị nhiễu bởi dữ liệu rác.

3.3.3 Kiểm tra và xử lý thiếu nhãn bộ/ngành

Trong bộ dữ liệu sau khi đã được lọc thông tin đầy đủ (`preprocess_2.csv`), vẫn còn một số bản ghi bị thiếu trường `ministry` – đại diện cho thông tin bộ/ngành quản lý. Đây là metadata quan trọng nhằm phân tích và đánh giá mô hình theo từng lĩnh vực, do vậy cần xử lý trước khi sử dụng trong quá trình sinh dữ liệu ảo giác.

a) Dữ liệu đầu vào

- Tệp sử dụng: `preprocess_2.csv`.
- Các trường thông tin cần chú ý: câu hỏi, câu trả lời đúng, bộ/ngành,...
- Số lượng bản ghi: 9205 mẫu.

b) Mục tiêu xử lý

- Tách các bản ghi bị thiếu thông tin bộ/ngành (cột `ministry` trống hoặc chứa chuỗi không hợp lệ).
- Làm sạch và chuẩn hóa tên bộ/ngành.
- Xuất tập lỗi riêng để xử lý bổ sung bằng tay nếu cần.

c) Phương pháp xử lý

1. Phân tích cột `ministry`:
    - Loại bỏ khoảng trắng thừa.
    - Gộp các biến thể cùng nghĩa (ví dụ: “Công an”, “Bộ Công An”, “CA” $\rightarrow$ “Bộ Công An”).

- Viết hoa chữ cái đầu, chuẩn hóa tên bộ/ngành theo danh sách có sẵn.

2. Xác định các dòng có giá trị trống, "null", hoặc không xác định trong cột `ministry` để xuất thành file riêng: `missing_ministry.csv`, từ đó có thể giữ lại cho xử lý thủ công nếu cần.

3. Với phần dữ liệu còn lại: đảm bảo 100% bản ghi có nhãn bộ/ngành hợp lệ.

4. Lưu kết quả ra file chính thức cho các bước tiếp theo.

d) Kết quả đầu ra

- Tệp trả về: `preprocess_3.csv` (chứa các bản ghi đầy đủ thông tin bộ/ngành).
- Số lượng bản ghi tương ứng: 3717 mẫu.
- Tệp trả về: `missing_ministry.csv` (các bản ghi thiếu thông tin để xử lý thủ công).
- Số lượng bản ghi tương ứng: 2358 mẫu.
- Tệp `preprocess_3.csv` được sử dụng làm chuẩn đầu vào cho chương tiếp theo, và được sao lưu dưới tên: `preprocessed_link.csv`.

e) Nhận xét

Việc đảm bảo tất cả các câu hỏi được gắn nhãn bộ/ngành giúp mở rộng khả năng phân tích mô hình theo từng lĩnh vực cụ thể, hỗ trợ đánh giá định hướng ngành, phân tích lỗi theo miền tri thức và phục vụ mục tiêu fine-tune mô hình chuyên biệt sau này. Đây là bước tiền xử lý quan trọng cuối cùng đối với tập câu hỏi – câu trả lời trước khi đưa vào giai đoạn sinh dữ liệu ảo giác.

3.3.4 Tiền xử lý dữ liệu tri thức (TTHC)

Bên cạnh việc xử lý tập câu hỏi – câu trả lời, hệ thống còn cần xử lý tập dữ liệu tri thức nền là các thủ tục hành chính (TTHC), vốn đóng vai trò quan trọng trong việc sinh và đánh giá các phản hồi có hoặc không có ảo giác. TTHC chính là căn cứ pháp lý để mô hình kiểm tra độ đúng sai trong nội dung sinh ra.

Tuy nhiên, dữ liệu TTHC được crawl từ nhiều trang khác nhau trên **Cổng Dịch vụ công Quốc gia**, có cấu trúc HTML đa dạng, không đồng nhất, và thường bao gồm nhiều mục thông tin rải rác như: tên thủ tục, cơ quan thực hiện, điều kiện, thành phần hồ sơ, thời hạn, cơ sở pháp lý,... Vì vậy, cần tổng hợp và chuẩn hóa lại tập dữ liệu này để phù hợp cho các bước sinh prompt và đánh giá mô hình. Toàn bộ quy trình này được thực hiện trong notebook `preprocess_tthc.ipynb`.

a) Dữ liệu đầu vào

- Tệp sử dụng: `raw_tthc.csv`.
- Các trường thông tin cần chú ý: mã thủ tục, tên thủ tục, đối tượng thực hiện, cách thức thực hiện,...
- Số lượng bản ghi: 2695 mẫu.

b) Mục tiêu xử lý

- Đồng bộ hóa tập TTHC với tập câu hỏi, chỉ giữ lại các TTHC thực sự được tham chiếu trong `preprocessed_link.csv`.
- Loại bỏ các bản ghi không có trong danh sách liên kết.
- Chuẩn hóa các giá trị rỗng hoặc thiếu thông tin về định dạng thống nhất.

c) Phương pháp thực hiện

1. Đọc toàn bộ tập dữ liệu `raw_tthc.csv`.

2. Trích xuất tập hợp các URL TTHC được sử dụng trong các câu hỏi.

3. Lọc tập `raw_tthc.csv` để chỉ giữ lại các dòng có `tthc_url` nằm trong tập URL liên kết.

4. Thay thế các giá trị thiếu (NaN) trong các trường thông tin chi tiết bằng chuỗi rỗng hoặc `"Không có thông tin"`.

d) Kết quả đầu ra

- Tệp trả về: `preprocessed_tthc.csv`.
- Số lượng bản ghi: 1820 mẫu.

e) Nhận xét

Tập dữ liệu kiến thức liên quan đã chuẩn hóa, bao gồm toàn bộ thông tin chi tiết của các thủ tục hành chính có liên kết thực tế trong tập câu hỏi. Tập tri thức TTHC đóng vai trò như một nguồn tham chiếu pháp lý, là cơ sở để xác định phản hồi nào là đúng hoặc có chứa ảo giác. Việc lọc và đồng bộ tập TTHC với các URL liên kết từ tập câu hỏi giúp đảm bảo rằng mọi phản hồi đều có tri thức đi kèm. Dữ liệu kết quả sau bước này là cơ sở chính thức để sử dụng trong các bước sinh và đánh giá phản hồi ở các chương tiếp theo.

Mô tả bộ dữ liệu sau khi tiền xử lý

- Tập dữ liệu câu hỏi - câu trả lời với số lượng bản ghi còn lại là: 3717 mẫu.
- Tập dữ liệu kiến thức liên quan (TTHC) với số lượng bản ghi còn lại là: 1820 mẫu.

Các thông tin mô tả chi tiết về bộ dữ liệu (giải thích các trường dữ liệu) được trình bày ở phần phụ lục A.

3.4 Kiểm tra dữ liệu thủ công

Sau khi hoàn tất giai đoạn tiền xử lý, dữ liệu vẫn cần được rà soát thủ công nhằm đảm bảo độ chính xác, đặc biệt là về chính tả và định dạng. Việc kiểm tra được thực hiện thông qua giao diện web do nhóm phát triển bằng Streamlit, chia thành hai phần: dữ liệu câu hỏi – câu trả lời (link) và dữ liệu thủ tục hành chính (TTHC). Các giao diện (Streamlit) dành cho annotator liên quan đến phần kiểm tra chính tả và chú thích ảo giác trong bộ dữ liệu được trình bày ở phần phụ lục B.

3.4.1 Kiểm tra dữ liệu câu hỏi – câu trả lời (link)

a) Phạm vi dữ liệu

Bộ dữ liệu "link" bao gồm các cặp câu hỏi và câu trả lời liên quan đến dịch vụ công. Dữ liệu được chia làm hai phần:

- `first_link.csv`: nửa đầu tập dữ liệu, do annotator 1 phụ trách kiểm tra.
- `second_link.csv`: nửa sau tập dữ liệu, do annotator 2 phụ trách kiểm tra.

b) Mục tiêu kiểm tra

- Rà soát và sửa lỗi chính tả trong nội dung câu hỏi và câu trả lời. Ngoài ra còn chú ý đến những lỗi định dạng như: ký tự xuống dòng, các ký tự gạch đầu dòng, đánh số mục,...
- Không thay đổi ngữ nghĩa hoặc biên tập lại văn bản.
- Kiểm tra tính hợp lệ của các trường đi kèm:
  - Phân loại người dùng (công dân, doanh nghiệp, tổ chức).
  - Thông tin bộ/ngành quản lý.
  - Danh sách thủ tục hành chính liên quan.
  - Các câu hỏi liên quan (nếu có).

c) Giao diện kiểm tra

Giao diện kiểm tra được xây dựng bằng Streamlit, hỗ trợ người dùng:

- Chọn bản ghi cần kiểm tra thông qua thanh trượt hoặc nhập số dòng.
- Xem đầy đủ thông tin gốc của bản ghi, bao gồm:
  - Liên kết gốc (URL).
  - Bộ/ngành chủ quản.
  - Đối tượng người dùng.
  - Danh sách các thủ tục hành chính liên quan.
  - Các câu hỏi liên quan.
- Chỉnh sửa nội dung câu hỏi và câu trả lời trong các ô nhập liệu.
- Xem gợi ý sửa lỗi chính tả từ mô hình Gemini (không áp dụng tự động).
- Đánh dấu bản ghi đã kiểm tra và lưu lại thời điểm chỉnh sửa.

d) Ghi nhận và lưu kết quả

- Mỗi bản ghi được đánh dấu là đã kiểm tra hoặc chưa kiểm tra.
- Thời gian cập nhật được lưu tự động mỗi khi người dùng xác nhận hoặc chỉnh sửa.
- Dữ liệu sau khi chỉnh sửa được lưu vào các file gốc: `annotated_data/first_link.csv` và `annotated_data/second_link.csv` phục vụ cho bước hậu xử lý sau này.

e) Lưu ý kỹ thuật

- Khi hết quota của API Gemini, cần thay đổi khóa truy cập trong file cấu hình.
- Dữ liệu gợi ý từ AI chỉ mang tính chất tham khảo, không được áp dụng tự động.

- Có thể bỏ qua các cảnh báo giao diện liên quan đến widget hoặc trạng thái phiên làm việc.

3.4.2 Kiểm tra dữ liệu thủ tục hành chính (TTHC)

a) Phạm vi dữ liệu

Dữ liệu liên quan đến thủ tục hành chính (TTHC) bao gồm nhiều trường mô tả mang tính định danh và nghiệp vụ, như tên thủ tục, điều kiện thực hiện, thành phần hồ sơ, thời gian giải quyết, cơ quan giải quyết, v.v. Tập dữ liệu được chia làm hai phần riêng biệt:

- `first_tthc.csv`: nửa đầu của tập dữ liệu, do annotator 1 phụ trách kiểm tra.
- `second_tthc.csv`: nửa sau của tập dữ liệu, do annotator 2 phụ trách kiểm tra.

b) Mục tiêu kiểm tra

- Kiểm tra chính tả cho tất cả các trường văn bản trong mỗi thủ tục hành chính.
- Không thay đổi từ ngữ, viết lại nội dung hoặc đưa ra diễn giải thêm.
- Chỉ sửa các lỗi chính tả hoặc ký tự đặc biệt sai định dạng.

Do đặc thù dữ liệu mang tính pháp lý và nghiệp vụ, việc bảo toàn nội dung gốc là bắt buộc.

c) Giao diện kiểm tra

Giao diện kiểm tra hỗ trợ trực quan, cho phép:

- Chọn bản ghi theo chỉ số để xem chi tiết từng thủ tục.
- Hiển thị các trường thông tin chính như:
  - Mã thủ tục (`maThuTuc`).

  - Đường dẫn gốc đến thủ tục hành chính.
  - Ngày cập nhật gần nhất.
  - Trạng thái kiểm tra.

- Với từng trường, người dùng có thể:
  - Xem nội dung ban đầu.
  - Sửa trực tiếp trong khung nhập liệu.
  - Nhận đề xuất sửa chính tả từ mô hình Gemini.
  - Lưu lại nội dung đã chỉnh sửa và cập nhật trạng thái kiểm tra.

d) Ghi nhận và lưu kết quả

- Cho phép khôi phục nội dung gốc nếu chỉnh sửa không phù hợp.
- Tự động lưu sau mỗi chỉnh sửa hoặc khi đánh dấu đã kiểm tra.
- Dữ liệu được ghi đè trực tiếp vào các file tại thư mục `annotated_data/`.
- Trường thời gian chỉnh sửa gần nhất được cập nhật tự động, phục vụ việc truy vết.

e) Lưu ý kỹ thuật

- Các trường có dung lượng lớn cần được kiểm tra cẩn thận, tránh thao tác vội vàng.
- Nếu mô hình Gemini trả về lỗi (do quota hoặc timeout), có thể nhấn lại nút đề xuất nhiều lần.
- Các trường không có nội dung thực chất (ví dụ: "Không có thông tin") có thể được bỏ qua.
- Các cảnh báo giao diện (liên quan đến session, widget...) không ảnh hưởng đến kết quả và có thể bỏ qua.

3.4.3 Tổng hợp sau kiểm tra và xuất dữ liệu

a) Mục tiêu tổng hợp

Sau khi hoàn tất quá trình kiểm tra chính tả thủ công trên toàn bộ bốn tập dữ liệu (`first_link.csv`, `second_link.csv`, `first_tthc.csv`, `second_tthc.csv`), bước tiếp theo là tiến hành tổng hợp dữ liệu. Mục tiêu chính là:

- Gộp các phần đã kiểm tra từ mỗi annotator thành hai bảng hoàn chỉnh: câu hỏi – câu trả lời (`link`) và thủ tục hành chính (`tthc`).
- Chỉ giữ lại các bản ghi có trường `checked=True`.
- Chuẩn hóa kiểu dữ liệu (đặc biệt với các cột chứa list như `tthc_list`).

b) Công cụ thực hiện

Toàn bộ bước tổng hợp được thực hiện trong notebook `postprocess.ipynb`, bao gồm:

- Đọc dữ liệu từ bốn file gốc đã kiểm duyệt.
- Ghép nối dữ liệu từ mỗi annotator.
- Lọc các bản ghi chưa được đánh dấu kiểm tra.
- Xử lý định dạng danh sách và chuỗi ký tự đặc biệt.
- Ghi kết quả cuối cùng ra thư mục chuẩn.

c) Kết quả đầu ra

- `postprocessed_link.csv`: chứa toàn bộ các câu hỏi và câu trả lời đã kiểm duyệt thủ công, có số lượng bản ghi không đổi là 3717 mẫu.
- `postprocessed_tthc.csv`: chứa toàn bộ các thủ tục hành chính đã kiểm duyệt thủ công, có số lượng bản ghi không đổi là 1820 mẫu.

Dữ liệu đầu ra là kết quả cuối cùng sau toàn bộ quá trình thu thập, xử lý và kiểm tra, sẵn sàng phục vụ cho giai đoạn sinh dữ liệu ảo giác và đánh giá mô hình.

3.5 Sinh dữ liệu ảo giác

Trong bối cảnh xây dựng hệ thống hỏi–đáp cho dịch vụ công, việc phát hiện và xử lý các câu trả lời chứa ảo giác, tức là nội dung không đúng hoặc không có trong nguồn thông tin gốc, là một bước quan trọng. Để huấn luyện và đánh giá các mô hình có khả năng nhận biết ảo giác, nhóm thực hiện quy trình sinh dữ liệu có kiểm soát, với hai bước chính tương ứng hai notebook.

3.5.1 Sinh dữ liệu ảo giác bằng mô hình ngôn ngữ

a) Mục tiêu

Giai đoạn này nhằm tạo ra các phản hồi chứa lỗi ảo giác (hallucination) từ mô hình ngôn ngữ lớn, phục vụ cho việc xây dựng bộ dữ liệu huấn luyện và đánh giá mô hình phát hiện ảo giác trong hệ thống hỏi–đáp dịch vụ công. Mỗi phản hồi được tạo ra phải đảm bảo tính trôi chảy ngôn ngữ, giữ được ngữ cảnh, nhưng chứa nội dung sai lệch hoặc không được hỗ trợ bởi tri thức liên quan.

b) Lựa chọn mô hình và siêu tham số

Chúng tôi sử dụng mô hình GPT-4o-mini do OpenAI phát triển, với ưu điểm về chi phí API hợp lý và khả năng sinh ngôn ngữ tự nhiên ổn định. Mặc dù có nhiều lựa chọn mô hình khác như Claude, Gemini hoặc DeepSeek, song GPT-4o-mini đáp ứng đủ yêu cầu trong tác vụ sinh văn bản hành chính cơ bản, không yêu cầu suy luận nâng cao.

Ba siêu tham số chính được sử dụng trong quá trình sinh:

- `temperature=1`: tăng độ ngẫu nhiên trong việc chọn từ tiếp theo.
- `top_p=1`: mở rộng không gian lựa chọn từ tiếp theo dựa trên xác suất tích lũy.
- `max_output_tokens=512`: đảm bảo độ dài phản hồi đủ lớn để diễn đạt được ảo giác.

c) Thiết kế truy vấn sinh ảo giác

Để hướng dẫn mô hình sinh ra phản hồi ảo giác đúng mục tiêu, chúng tôi xây dựng truy vấn với cấu trúc rõ ràng gồm hai phần:

- **System prompt**: định nghĩa vai trò mô hình là một trình tạo phản hồi ảo giác, mô tả mục tiêu và ràng buộc độ dài.
- **User prompt**: cung cấp ngữ cảnh gồm tri thức liên quan, câu hỏi gốc và câu trả lời đúng.

Truy vấn được thiết kế như sau:

Bảng 3.1: Truy vấn sinh ảo giác

Truy vấn hệ thống:
Bạn sẽ đóng vai trò là một trình tạo câu trả lời ảo giác (hallucination answer generator). Với một câu hỏi, câu trả lời đúng, và kiến thức liên quan, mục tiêu của bạn là viết một câu trả lời ảo giác mà nghe có về đúng nhưng thực tế lại sai. *{pattern}* Bạn nên cố gắng hết sức để làm cho câu trả lời trở nên ảo giác. #Câu trả lời ảo giác# chỉ có thể nhiều hơn #Câu trả lời đúng# khoảng 5 từ.
Truy vấn người dùng (học ngữ cảnh không có ví dụ):
#Kiến thức liên quan#: *{knowledge}* #Câu hỏi#: *{question}* #Câu trả lời đúng#: *{right_ answer}* #Câu trả lời ảo giác#:

d) Phân loại ảo giác (Pattern)

Nhằm đa dạng hóa các kiểu lỗi sinh ra và phục vụ cho việc phân tích sau này, chúng tôi định nghĩa bốn dạng ảo giác phổ biến (pattern), mỗi phản hồi được gán ngẫu nhiên một dạng:

1. Hiểu sai ngữ cảnh hoặc mục đích câu hỏi.

2. Mâu thuẫn giữa câu trả lời và tri thức liên quan.

3. Trả lời quá chung chung hoặc quá chi tiết.

4. Suy luận sai từ tri thức có sẵn.

Mỗi pattern được nhúng vào `system_prompt` để điều hướng hành vi sinh nội dung của mô hình.

e) Quy trình thực hiện

Toàn bộ quy trình được triển khai trong notebook `hallucination_generate_gpt.ipynb`, với các bước chính:

1. Đọc dữ liệu đầu vào gồm:

   - `postprocessed_link.csv`: chứa câu hỏi, câu trả lời và các thủ tục hành chính liên quan.
   - `postprocessed_tthc.csv`: chứa nội dung các thủ tục hành chính liên quan.

2. Với mỗi bản ghi:

   - Trích xuất tri thức liên quan từ các URL trong `TTHCLienQuan`.
   - Xây dựng `system_prompt` và `user_prompt`.
   - Gọi mô hình GPT-4o-mini để sinh phản hồi ảo giác.
   - Lưu kết quả vào file `hallucination_generate_gpt.csv` gồm: `link`, `cauTraLoiAoGiac`, `pattern`.

f) Kết quả đầu ra

- Tệp trả về: `hallucination_generate_gpt.csv`
- Các trường thông tin cần chú ý:
  - `link`: URL câu hỏi ban đầu.
  - `cauTraLoiAoGiac`: phản hồi sinh ra bởi GPT, có tính ảo giác.

  - `pattern`: chỉ số kiểu ảo giác áp dụng (0–3), tương ứng theo thứ tự phân loại pattern từ I đến IV.

- Số lượng bản ghi: 3717 mẫu.

3.5.2 Hậu xử lý dữ liệu ảo giác

a) Mục tiêu

Sau khi sinh phản hồi ảo giác từ mô hình GPT-4o-mini, bước tiếp theo là tiến hành hậu xử lý dữ liệu nhằm đảm bảo:

- Tập dữ liệu đầu ra không chứa lỗi kỹ thuật hoặc định dạng.
- Loại bỏ các phản hồi không hợp lệ hoặc không mang tính ảo giác.
- Chuẩn hóa văn bản và nhãn, sẵn sàng phục vụ huấn luyện và đánh giá mô hình.

b) Quy trình thực hiện

Toàn bộ quy trình được thực hiện trong notebook `postgenerate_gpt.ipynb`, với các bước cụ thể như sau:

1. **Nạp dữ liệu đầu vào:**
   - Tập dữ liệu gốc: `postprocessed_link.csv`.
   - Tập phản hồi ảo giác: `hallucination_generate_gpt.csv`.
   - Kết hợp hai nguồn dữ liệu dựa trên `link`, bổ sung hai trường:
     - `cauTraLoiAoGiac`: phản hồi ảo giác sinh ra.
     - `pattern`: phân loại dạng ảo giác tương ứng.
2. **Làm sạch nội dung phản hồi:**
   - Loại bỏ các tiền tố không cần thiết như "Câu trả lời ảo giác: ...".
   - Gỡ bỏ các chú thích hệ thống do mô hình sinh ra (ví dụ: cảnh báo hoặc giải thích).

- Chuẩn hóa ký tự đầu dòng, khoảng trắng và định dạng Markdown.

3. **Phát hiện lỗi sinh:**

   - Lọc các phản hồi có chứa từ khoá không mong muốn như "Câu hỏi", "Câu trả lời" (lỗi định dạng prompt).
   - Tổng hợp số lượng lỗi để xử lý thủ công nếu cần thiết.

4. **Kiểm tra phân phối pattern:**

   Sau khi tổng hợp, tập dữ liệu được phân chia khá đều giữa bốn dạng ảo giác, cụ thể:

   - Loại I (hiểu sai ngữ cảnh hoặc mục đích của câu hỏi): 922 bản ghi.
   - Loại II (mâu thuẫn với tri thức liên quan): 915 bản ghi.
   - Loại III (trả lời quá chung hoặc quá chi tiết): 941 bản ghi.
   - Loại IV (suy luận sai từ tri thức): 939 bản ghi.

   Phân phối đều giữa các loại phản hồi giúp bộ dữ liệu đạt được độ đa dạng cần thiết và thuận lợi cho phân tích sau này.

5. **Lưu kết quả:**

   - Tập dữ liệu hoàn chỉnh được lưu dưới tên: `postgenerate_gpt.csv`.
   - Bao gồm 9 trường chính: `link`, `phanLoai`, `boNganh`, `cauHoi`, `cauTraLoi`, `TTHCLienQuan`, `cauHoiLienQuan`, `cauTraLoiAoGiac`, `pattern`.
   - Kích thước tập dữ liệu sau hậu xử lý: 3717 dòng.

c) Nhận xét

Bước hậu xử lý giúp đảm bảo toàn bộ tập dữ liệu phản hồi ảo giác không chỉ đúng về hình thức mà còn đạt yêu cầu kỹ thuật để phục vụ các tác vụ downstream như:

- Huấn luyện mô hình phân loại phản hồi thật/ảo giác.
- Kiểm định khả năng phát hiện lỗi nội dung của LLM.
- Làm tập kiểm thử cho hệ thống hỏi – đáp pháp lý.

Việc giữ lại cả thông tin gốc (câu hỏi, câu trả lời đúng, bộ/ngành, danh sách tri thức liên quan) cùng phản hồi ảo giác sẽ là cơ sở quan trọng để đánh giá chất lượng phát hiện ảo giác trong các nghiên cứu tiếp theo.

3.6 Chú thích dữ liệu ảo giác

Sau khi sinh phản hồi từ mô hình ngôn ngữ, cần có bước đánh giá thủ công để xác định phản hồi nào thực sự chứa ảo giác, nhằm đảm bảo chất lượng tập dữ liệu gán nhãn. Giai đoạn này đóng vai trò quan trọng trong việc xây dựng tiêu chuẩn (benchmark) tin cậy, phục vụ huấn luyện và đánh giá mô hình phát hiện ảo giác. Quá trình được thực hiện qua một hệ thống web chú thích, với sự tham gia của hai annotator độc lập, có đối chiếu kết quả nhằm đảm bảo tính đồng thuận và khách quan.

a) Mục tiêu

Sau khi sinh ra các phản hồi ảo giác từ mô hình ngôn ngữ lớn (xem Mục 3.5), bước tiếp theo là tiến hành chú thích thủ công nhằm xác định:

- Liệu phản hồi được sinh ra có thực sự là ảo giác hay không.
- Nếu có, phản hồi đó có đúng dạng ảo giác đã định nghĩa (pattern) hay không.

Mục tiêu của quá trình này là đảm bảo độ chính xác cao cho bộ dữ liệu, loại bỏ các mẫu không đạt yêu cầu, từ đó tăng độ tin cậy cho quá trình huấn luyện và đánh giá mô hình phát hiện ảo giác.

b) Chuẩn bị dữ liệu đầu vào

Trước khi tiến hành gán nhãn thủ công, dữ liệu được xử lý sơ bộ từ tập `postgenerate_gpt.csv`. Bốn trường mới được thêm vào gồm:

- `checked`: xác định mẫu đã được con người kiểm tra hay chưa (bool).
- `lastUpdated`: thời điểm kiểm tra cuối cùng (timestamp).
- `hallucinated`: xác định mẫu có phải là ảo giác hay không (bool).
- `rightPattern`: xác định pattern được mô hình gán có đúng hay không (bool).

Sau đó, dữ liệu được nhân bản thành hai tập riêng biệt: `human1.csv` và `human2.csv`, phục vụ hai người gán nhãn độc lập nhằm đảm bảo tính khách quan.

c) Giao diện gán nhãn

Giao diện chú thích được xây dựng bằng thư viện `Streamlit`, chia thành ba tab chính:

- First human: tương tác với `human1.csv`.
- Second human: tương tác với `human2.csv`.
- Recheck: dự kiến dùng để xử lý bất đồng (chưa triển khai).

Với mỗi dòng dữ liệu, annotator được cung cấp các thông tin sau:

- Câu hỏi, câu trả lời gốc và câu trả lời ảo giác.
- Dạng ảo giác (pattern) mà mô hình đã gán.
- Danh sách thủ tục hành chính liên quan và các câu hỏi liên quan (nếu có).
- Các trường tương tác: đánh dấu là ảo giác thật, sai pattern, không ảo giác, chưa kiểm tra.

Các lựa chọn gán nhãn bao gồm:

- Unchecked: chưa xác nhận mẫu.

- False hallucination: phản hồi không phải ảo giác.
- Only false pattern: phản hồi là ảo giác nhưng sai loại.
- True hallucination: phản hồi đúng là ảo giác và đúng pattern.

d) Tiêu chí chú thích

Annotator được hướng dẫn sử dụng các tiêu chí nghiêm ngặt để xác định phản hồi ảo giác, bao gồm:

- Sai lệch về số liệu, thời gian, điều kiện thủ tục hoặc tên cơ quan.
- Mâu thuẫn với tri thức gốc (trích từ thủ tục hành chính liên quan).
- Bịa đặt thông tin không xuất hiện trong tri thức đã cung cấp.
- Trả lời quá chung chung hoặc không trả lời đúng vào trọng tâm câu hỏi.

Định nghĩa các loại ảo giác được sử dụng nhất quán như sau:

1. Pattern loại I: Hiểu sai ngữ cảnh hoặc mục đích của câu hỏi.
2. Pattern loại II: Mâu thuẫn với kiến thức liên quan.
3. Pattern loại III: Trả lời quá chung hoặc quá chi tiết.
4. Pattern loại IV: Suy luận sai từ kiến thức có sẵn.

e) Kết quả và lưu trữ

Sau quá trình chú thích:

- Mỗi annotator đã thực hiện gán nhãn độc lập trên toàn bộ 3717 phản hồi.
- Các lựa chọn được ghi nhận trực tiếp vào các cột tương ứng trong tệp `human1.csv` và `human2.csv`.
- Mỗi lần cập nhật đều lưu lại thời gian và trạng thái mới của mẫu.

Việc đồng gán nhãn từ hai annotator giúp tăng độ tin cậy và cho phép xử lý bất đồng (nếu cần) bằng một vòng gán nhãn lại hoặc trung gian.

f) Kết luận

Quá trình gán nhãn thủ công là một bước không thể thiếu để đảm bảo chất lượng và độ chính xác cho bộ dữ liệu ảo giác. Nhờ kết hợp giữa sinh tự động và kiểm tra thủ công, tập dữ liệu đầu ra có thể được sử dụng làm benchmark đáng tin cậy cho các nghiên cứu phát hiện lỗi nội dung và kiểm soát ảo giác của mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công.

Lưu ý: Bước chú thích dữ liệu ảo giác này *xem như một bước bổ sung* nếu như nguồn dữ liệu được chọn đã quá tốt, hay giai đoạn tiền xử lý dữ liệu đầu vào cho mô hình đã đảm bảo chất lượng, hay mô hình ngôn ngữ lớn được chọn để sinh dữ liệu ảo giác đã rất mạnh (như mô hình GPT-4o). Trong tình huống này, các câu trả lời được sinh ra hầu như đã đảm bảo 100% tính ảo giác khi thực hiện quan sát tổng quan dữ liệu, vì vậy nên bước chú thích dữ liệu ảo giác này sẽ không thực sự cần thiết.

3.7 Mô tả bộ dữ liệu cuối cùng

Bảng 3.2 tổng hợp kích thước hai tập dữ liệu chính, gồm 7434 mẫu câu hỏi và 1820 thủ tục hành chính được tham chiếu. Bảng 3.4 cho thấy phân bố mẫu theo bộ/ngành, phản ánh mức độ phổ biến của từng lĩnh vực trong dữ liệu. Bảng 3.3 trình bày phân bố 3717 mẫu dương theo bốn phân loại ảo giác, với tỷ lệ tương đối đồng đều giữa các nhóm. Ngoài ra, các thông tin mô tả chi tiết về bộ dữ liệu (giải thích các trường dữ liệu) được trình bày ở phần phụ lục A.

Bảng 3.2: Tổng quan các tập dữ liệu

Tập dữ liệu câu hỏi

	Số lượng
Số câu hỏi	3717
Số câu trả lời đúng (mẫu âm)	3717
Số câu trả lời ảo giác (mẫu dương)	3717
Tổng số mẫu	**7434**
Số mẫu dương	3717
Số mẫu âm	3717
Số mẫu sử dụng khi không truyền kiến thức	**7434**
Số mẫu sử dụng khi có truyền kiến thức	2000

Tập dữ liệu kiến thức liên quan

	Số lượng
Tổng số TTHC	1820
Số TTHC được tham chiếu	1820

Bảng 3.3: Phân bố dữ liệu trên thuộc tính phân loại ảo giác (pattern) (chỉ xét ở những mẫu dương)

Pattern	**Mô tả**	**Số mẫu**	**Tỷ lệ (%)**
P-I	Hiểu sai ngữ cảnh và mục đích	922	24.80
P-II	Mâu thuẫn giữa câu trả lời và tri thức	915	24.62
P-III	Quá chung chung hoặc quá chi tiết	941	25.32
P-IV	Suy luận sai từ tri thức	939	25.26
Tổng cộng		**3717**	**100**

Bảng 3.4: Phân bố dữ liệu trên thuộc tính bộ/ngành

Bộ/ngành	**Số mẫu**	**Tỷ lệ (%)**
Bộ Nông nghiệp và Môi trường	1174	15.79
Bộ Giao thông vận tải	900	12.11
Bộ Khoa học và Công nghệ	808	10.87
Bộ Tư pháp	738	9.93
Bộ Công an	680	9.15
Bộ Quốc phòng	620	8.34
Bộ Ngoại giao	556	7.48
Bộ Y tế	526	7.08
Bộ Nội vụ	416	5.60
Bộ Tài chính	292	3.93
Thanh tra Chính phủ	260	3.50
Bộ Tài nguyên và Môi trường	248	3.34
Bộ Công Thương	126	1.69
Bộ Lao động - Thương binh và Xã hội	90	1.21
Tổng cộng	**7434**	**100**

Chương 4

Đánh giá trên các mô hình

4.1 Tổng quan quy trình đánh giá các mô hình ngôn ngữ lớn

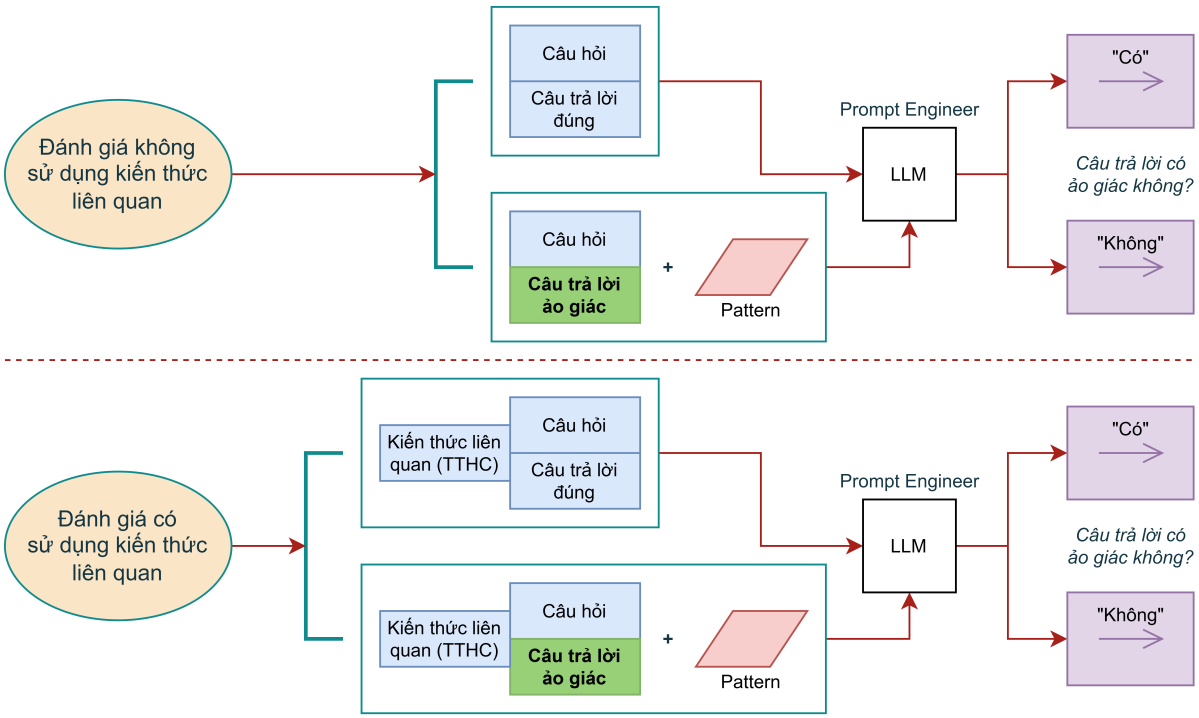
Hình 4.1 mô tả quy trình đánh giá khả năng phát hiện thông tin ảo giác của mô hình ngôn ngữ lớn (LLM). Quy trình gồm hai thiết lập bên dưới, cả hai thiết lập đều hướng đến việc kiểm định: LLM có thể nhận diện câu trả lời ảo giác trong ngữ cảnh dịch vụ công, với hoặc không với tri thức liên quan.

1. Đánh giá không sử dụng kiến thức liên quan:

- Mỗi mẫu gồm câu hỏi và hai câu trả lời: một đúng và một chứa thông tin ảo giác (được tạo theo từng pattern).
- Các thành phần này được đưa vào bộ khung prompt đánh giá.
- LLM đóng vai trò là người đánh giá, trả lời "Có" nếu phát hiện ảo giác, "Không" nếu không phát hiện.
- Mục tiêu là kiểm tra xem LLM có thể tự đánh giá tính đúng/sai của câu trả lời chỉ dựa trên bản thân nội dung câu hỏi và câu trả lời hay không.

2. Đánh giá có sử dụng kiến thức liên quan (TTHC):

- Mỗi mẫu được mở rộng thêm kiến thức liên quan, tức là toàn bộ thông tin từ thủ tục hành chính (TTHC) gắn với câu hỏi.
- Prompt lúc này chứa kiến thức + câu hỏi + câu trả lời (đúng hoặc ảo giác).



Hình 4.1: Quy trình đánh giá các mô hình ngôn ngữ lớn trong ngữ cảnh dịch vụ công

- Quá trình đánh giá được thực hiện tương tự: LLM đưa ra đánh giá "Có" hoặc "Không".
- Thiết lập này kiểm tra khả năng sử dụng tri thức nền khi đánh giá của mô hình.

4.2 Lựa chọn các mô hình sử dụng

Trong thực nghiệm này, chúng tôi đánh giá khả năng phát hiện thông tin ảo giác (hallucinated information) trong ngữ cảnh dịch vụ công của các mô hình ngôn ngữ lớn (LLM). Hai nhóm mô hình được lựa chọn gồm mô hình mã nguồn mở triển khai cục bộ và mô hình mã nguồn đóng truy cập qua API. Việc lựa chọn mô hình nhằm đảm bảo sự đa dạng về kiến trúc, nền tảng huấn luyện, và khả năng hỗ trợ tiếng Việt trong các tác vụ đánh giá nhị phân.

Nhóm mô hình mã nguồn mở

Các mô hình được triển khai cục bộ ở định dạng GGUF, thực thi bằng LM Studio. Tất cả đều có quy mô 7B tham số, tối ưu cho khả năng xử lý song ngữ và hoạt động hiệu quả trên phần cứng hạn chế. Các mô hình gồm:

- **Llama-3** (Meta AI): Mô hình mới từ Meta thuộc dòng Llama-3, huấn luyện trên dữ liệu chất lượng cao, hỗ trợ đa ngữ. Phiên bản Q4_K_M cân bằng giữa dung lượng và độ chính xác, có khả năng phản hồi tiếng Việt với prompt rõ ràng.
- **Mistral-v0.3** (Mistral AI): Mô hình nhỏ gọn, phản hồi nhanh, tối ưu cho thiết bị cá nhân. Hiệu quả với truy vấn ngắn và đánh giá hàng loạt.
- **Vicuna-v1.5** (LMSys): Tinh chỉnh từ LLaMA, nổi bật với khả năng hội thoại tự nhiên. Phù hợp với đánh giá truy vấn ngắn dạng nhị phân.
- **Qwen-2.5** (Alibaba Cloud): Hỗ trợ đa ngôn ngữ, tập trung vào tiếng Trung và tiếng Anh. Tuy vậy, vẫn phản hồi tiếng Việt tốt khi dùng prompt chặt chẽ.
- **WizardLM-2** (Microsoft Research): Huấn luyện bằng kỹ thuật evolutionary instruction tuning, hiệu quả với truy vấn có logic hoặc suy luận đơn giản, nhạy với sai lệch ngữ nghĩa.

Đặc biệt, đề tài đánh giá thêm hai mô hình được tinh chỉnh trên bộ dữ liệu tiếng Việt.

- **Vistral** (UoNLP): Mô hình pretrain và fine-tune hoàn toàn bằng tiếng Việt, tập trung vào đối thoại, QA và truy vấn hành chính công. Là một trong số ít mô hình mã nguồn mở thuần Việt, phản hồi trôi chảy và chính xác với truy vấn tiếng Việt.
- **Qwen2.5-Viet**: Phiên bản tiếng Việt hóa từ Qwen2.5, fine-tune trên tập song ngữ Việt–Anh. Tối ưu cho truy vấn văn bản hành chính và nhiệm vụ nhận diện ảo giác chuyên ngành.

Tất cả mô hình mã nguồn mở đều hỗ trợ phản hồi dạng nhị phân ("Có"/"Không"), có khả năng truy cập nội bộ không cần Internet, đảm bảo tính nhất quán trong quy trình đánh giá trên toàn bộ tập dữ liệu.

Nhóm mô hình mã nguồn đóng

Đây là các mô hình thương mại hiện đại, truy cập thông qua API (OpenRouter). Các mô hình được lựa chọn đảm bảo phản hồi ổn định, có khả năng hỗ trợ tiếng Việt, và đại diện cho các kiến trúc tiên tiến từ các tổ chức phát triển LLM hàng đầu:

- **GPT-4o-mini** (OpenAI): Phiên bản nhẹ của GPT-4o, tốc độ phản hồi nhanh, phù hợp cho đánh giá lặp lại. Hỗ trợ tiếng Việt tốt và giữ được khả năng suy luận mạnh.
- **DeepSeek-v3-0324** (DeepSeek AI): Mô hình phát hành 2024, nổi bật với khả năng tổng hợp và suy luận logic. Phản hồi ổn định với truy vấn dạng yes/no, hỗ trợ song ngữ.
- **Gemini-2.0-flash** (Google DeepMind): Tối ưu cho tốc độ, hỗ trợ đa ngôn ngữ. Dù không chuyên sâu tiếng Việt, vẫn phản hồi tốt với prompt rõ ràng.
- **Claude-3.5-Haiku** (Anthropic): Mô hình sinh văn bản chất lượng cao, phản hồi có chiều sâu nhưng hơi bảo thủ trong truy vấn nhị phân. Hỗ trợ tiếng Việt còn hạn chế.

Các mô hình mã nguồn đóng được kiểm soát đầu vào bằng prompt thống nhất, sử dụng template dạng ràng buộc chặt chẽ, cấu hình tham số giống nhau (temperature, top_p, max_tokens). Kết quả phản hồi được xử lý hậu kỳ để trích xuất nhãn đánh giá, phục vụ xây dựng confusion matrix và tính toán các chỉ số đánh giá (precision, recall, F1-score, accuracy).

4.3 Lựa chọn các siêu tham số và cấu hình thực nghiệm

Để đảm bảo tính công bằng và đồng nhất trong đánh giá giữa các mô hình, chúng tôi thiết lập chung các siêu tham số quan trọng cho toàn bộ

quá trình đánh giá:

- **Temperature**: 0.7
- **Top-p** (nucleus sampling): 0.9
- **Max tokens**:
  - 32 tokens cho mô hình mã nguồn mở
  - 512 tokens cho mô hình mã nguồn đóng

Các giá trị trên được lựa chọn nhằm duy trì sự ổn định trong phản hồi, đồng thời cho phép mô hình có đủ khả năng sinh câu trả lời nhị phân "Có" hoặc "Không" một cách rõ ràng. Việc phân biệt độ dài đầu ra giữa hai nhóm mô hình phản ánh đặc điểm triển khai thực tế, khi mã nguồn mở sử dụng trực tiếp trong môi trường nội bộ với cấu hình giới hạn, còn mô hình mã nguồn đóng gọi qua API và xử lý đầu ra dài hơn.

Đối với mã nguồn mở, các mô hình được triển khai cục bộ bằng giao diện lập trình thông qua thư viện chuyên dụng. Mỗi truy vấn gửi vào bao gồm phần hướng dẫn (system message) và nội dung câu hỏi + câu trả lời (user message), kèm theo cấu hình trên.

Thiết kế truy vấn đánh giá

Để hướng dẫn mô hình thực hiện nhiệm vụ đánh giá thông tin ảo giác đúng mục tiêu, chúng tôi xây dựng truy vấn với cấu trúc rõ ràng gồm hai phần:

- **System prompt**: định nghĩa vai trò của mô hình là một người đánh giá (answer judge), mô tả nhiệm vụ và ràng buộc định dạng câu trả lời.
- **User prompt**: cung cấp ngữ cảnh gồm câu hỏi và câu trả lời cần đánh giá.

Truy vấn được thiết kế như sau:

Bảng 4.1: Truy vấn đánh giá (không sử dụng kiến thức liên quan)

Truy vấn hệ thống:
Bạn sẽ đóng vai trò là một người đánh giá câu trả lời (answer judge). Với một câu hỏi và câu trả lời, mục tiêu của bạn là xác định xem câu trả lời được cung cấp có chứa thông tin không đúng sự thật hoặc thông tin áo giác (hallucinated information) hay không. *{pattern}* Bạn nên cố gắng hết sức để xác định xem câu trả lời có chứa thông tin không đúng sự thật hoặc thông tin áo giác hay không. Câu trả lời bạn đưa ra bắt buộc CHỈ là "Có" hoặc "Không", và không giải thích gì thêm. Trả lời "Có" nếu câu trả lời chứa thông tin áo giác, trả lời "Không" nếu câu trả lời không chứa thông tin áo giác.
Truy vấn người dùng (học ngữ cảnh không có ví dụ):
#Câu hỏi#: *{question}* #Câu trả lời#: *{answer}* #Đánh giá của bạn#:

4.4 Đánh giá trên các mô hình ngôn ngữ lớn mã nguồn mở (open source)

Với các mô hình mã nguồn mở, gồm Llama-3, Mistral-v0.3, Qwen-2.5, Vicuna-v1.5, WizardLM-2, cùng hai mô hình tinh chỉnh hỗ trợ tiếng Việt là Vistral và Qwen-Viet, quá trình đánh giá được triển khai hoàn toàn cục bộ thông qua nền tảng tương thích như LM Studio.

Bước 1: Khởi tạo mô hình và cấu hình tham số

Mỗi mô hình được gọi thông qua hàm `lms.llm(model_name)`. Các tham số đánh giá sử dụng cho tất cả mô hình mã nguồn mở là:

- `temperature = 0.7`
- `top_p = 0.9`
- `max_tokens = 32`

Các giá trị này đảm bảo phản hồi ngắn gọn, đúng theo yêu cầu "Có" hoặc "Không".

Bước 2: Chuẩn bị truy vấn đánh giá

Từ tập dữ liệu gồm 3.717 mẫu, hai truy vấn được sinh ra cho mỗi mẫu - tương ứng với câu trả lời đúng và câu trả lời ảo giác. Các truy vấn được cấu trúc theo định dạng tương tự như ở Mục 4.2, dùng template đánh giá cho mã nguồn mở.

Bước 3: Gửi truy vấn và thu thập kết quả

Mỗi truy vấn được truyền vào mô hình qua lệnh `model.respond(...)` với cấu hình đã định. Mỗi phản hồi là một chuỗi văn bản, được ép kiểu về chuỗi để lưu trữ và xử lý sau.

Các nội dung thu thập gồm:

- `danhGiaDung`: phản hồi cho câu trả lời đúng
- `danhGiaAoGiac`: phản hồi cho câu trả lời chứa ảo giác

Bước 4: Chuẩn hóa đầu ra và lưu trữ

Phản hồi đầu ra được kiểm tra định dạng để đảm bảo chỉ chứa "Có" hoặc "Không". Các phản hồi khác (rỗng, lỗi, không rõ ràng) sẽ được xử lý riêng. Kết quả cuối cùng được lưu vào tập tin CSV để phục vụ bước phân tích chỉ số đánh giá.

4.5 Đánh giá trên các mô hình ngôn ngữ lớn mã nguồn đóng (closed source)

Đối với các mô hình mã nguồn đóng (hoặc chỉ truy cập qua API), gồm GPT-4o-mini, Gemini-2.0-flash, Claude-3.5-Haiku và DeepSeek-v3-0324, quá trình đánh giá được tự động hóa thông qua API với các truy vấn được tạo từ bộ dữ liệu chuẩn hóa.

Bước 1: Chuẩn bị dữ liệu

Tập dữ liệu đầu vào gồm 3.717 mẫu, mỗi mẫu chứa cặp câu hỏi và hai câu trả lời: một đúng và một chứa thông tin ảo giác. Mỗi mẫu được gán nhãn pattern (từ 0 đến 3) tương ứng với phân loại ảo giác cụ thể.

Bước 2: Sinh truy vấn đánh giá

Với mỗi mẫu, hai truy vấn đánh giá được tạo:

- Một truy vấn kiểm tra câu trả lời đúng (không chứa `{pattern}`)
- Một truy vấn kiểm tra câu trả lời ảo giác (có chèn `{pattern}` phù hợp)

Các truy vấn tuân theo cấu trúc system–user như đã trình bày ở Mục 4.2, được điều chỉnh thành prompt hoàn chỉnh trước khi gửi đến API.

Bước 3: Truy vấn API và thu thập phản hồi

Mỗi truy vấn được gửi đến mô hình thông qua API với thông số đã định (`temperature = 0.7`, `top_p = 0.9`, `max_tokens = 512`). Mô hình phải phản hồi duy nhất bằng "Có" hoặc "Không". Kết quả được trích xuất từ nội dung phản hồi và lưu vào cột `danhGiaDung` và `danhGiaAoGiac` tương ứng cho từng mẫu.

Bước 4: Chuẩn hóa đầu ra

Do phản hồi từ API thường chứa thêm phần định dạng hoặc mô tả không cần thiết, một hàm trích xuất đơn giản được áp dụng để tách phần nội dung sau `content='...'` làm nhãn đánh giá. Kết quả chuẩn hóa này dùng làm đầu vào cho bước tính toán chỉ số đánh giá.

4.6 Đánh giá sử dụng kiến thức liên quan

Trong thiết lập này, chúng tôi đánh giá khả năng nhận diện ảo giác của các mô hình mã nguồn đóng khi được cung cấp đầy đủ tri thức chuyên ngành trong prompt. Mỗi truy vấn được tạo bằng cách đưa toàn bộ nội

dung của thủ tục hành chính tương ứng vào phần `system_message`, nhằm mô phỏng tình huống lý tưởng khi mô hình có quyền truy cập đầy đủ vào kiến thức liên quan.

Các bước thực hiện tương tự như phần đánh giá không sử dụng kiến thức:

- Với mỗi mẫu dữ liệu, sinh hai truy vấn: một cho câu trả lời đúng và một cho câu trả lời ảo giác.
- Toàn bộ nội dung tri thức (được trích xuất từ văn bản TTHC) được chèn vào phần `system`, đi kèm với hướng dẫn đánh giá và `{pattern}` nếu có.
- Prompt đầu vào có dạng: `system` chứa hướng dẫn đánh giá; `user` gồm tri thức + cặp câu hỏi – câu trả lời.
- Phản hồi từ API được trích xuất như trước, giữ lại nhãn "Có" hoặc "Không".

Trong cấu hình này, tập dữ liệu đánh giá gồm 2.000 mẫu (1000 mẫu dương và 1000 mẫu âm), lấy đối xứng từ tập dữ liệu ban đầu để đảm bảo cân bằng. Các mô hình sử dụng gồm GPT-4o-mini, Gemini-2.0-flash, Claude-3.5-Haiku và DeepSeek-v3-0324. Các tham số đánh giá giữ nguyên như trước: `temperature = 0.7`, `top_p = 0.9`, `max_tokens = 512`.

Ngoài ra, nhận thấy được sự tiềm năng của những mô hình mã nguồn mở trong việc đánh giá ảo giác trong ngữ cảnh dịch vụ công (kết quả được thể hiện ở bảng 4.2), chúng tôi cũng đánh giá thêm 2 mô hình mã nguồn mở có độ chính xác (accuracy) tốt nhất, đó là WizardLM-2 và Qwen-Viet. Việc cài đặt tương tự như trên, đảm bảo sự công bằng về bộ dữ liệu và các siêu tham số chung.

Mẫu truy vấn được thiết kế cho các mô hình mã nguồn đóng trong việc đánh giá sử dụng kiến thức liên quan được trình bày ở phần phụ lục C.

4.7 Kết quả và nhận xét

Trong thiết lập không truyền kiến thức liên quan, 11 mô hình được đánh giá khả năng phát hiện thông tin ảo giác chỉ dựa trên câu hỏi và câu

trả lời. Bảng 4.2 cho thấy sự chênh lệch rõ rệt giữa các mô hình, dù độ chính xác nhìn chung dao động quanh mức 50%. Đáng chú ý, một số mô hình mã nguồn mở lại cho kết quả nổi bật hơn cả các mô hình mã nguồn đóng.

Mô hình WizardLM-2 đạt độ chính xác cao nhất (57.61%), vượt qua toàn bộ các mô hình thương mại, bao gồm GPT-4o-mini (51.72%) và Gemini-2.0-flash (51.29%). Bên cạnh đó, Qwen-Viet — một mô hình mã nguồn mở đã tinh chỉnh tiếng Việt — cũng đạt kết quả tốt (53.81%), khẳng định tiềm năng của việc địa phương hóa mô hình. Các mô hình khác như LLaMA-3 (53.07%), Qwen-2.5 (51.91%) hay Vicuna-v1.5 (50.59%) đều cho thấy độ chính xác tương đương hoặc vượt nhẹ so với các mô hình mã nguồn đóng.

Ngược lại, mô hình Claude-3.5-Haiku thể hiện kết quả yếu nhất (43.58%), cho thấy hạn chế trong việc đánh giá nội dung ảo giác khi thiếu kiến thức nền. Nhìn chung, các kết quả này cho thấy mô hình mã nguồn mở hoàn toàn có khả năng cạnh tranh và thậm chí vượt trội hơn so với mô hình thương mại khi đánh giá thông tin ảo giác trong ngữ cảnh dịch vụ công, ngay cả khi không sử dụng kiến thức liên quan.

Ngoài ra, thống kê chi tiết trên những độ đo khác như precision, recall, f1-score ở các mô hình ngôn ngữ lớn được trình bày ở phần phụ lục D.

Từ kết quả ở bảng 4.2, ta nhận thấy được sự nổi bật về độ chính xác của mô hình WizardLM-2. Từ đó, nhóm nghiên cứu quyết định lựa chọn mô hình này để đánh giá tiếp trên hai tiêu chí: bộ/ngành và phân loại ảo giác (pattern).

Bảng 4.3 trình bày độ chính xác của mô hình WizardLM-2 trong việc phát hiện thông tin ảo giác theo từng bộ/ngành. Khác với nhận xét trước đó khi đánh giá trung bình nhiều mô hình, kết quả ở đây thể hiện rõ mức độ phân hóa giữa các lĩnh vực cụ thể. Một số bộ/ngành đạt độ chính xác nổi bật, có thể kể đến như Bộ Lao động – Thương binh và Xã hội (64.44%), Bộ Quốc phòng (62.58%), Bộ Y tế (58.56%) và Bộ Tư pháp (58.94%). Những kết quả này cho thấy mô hình WizardLM-2 có khả năng nhận diện thông tin ảo giác tốt hơn trong các ngữ cảnh pháp lý giàu cấu trúc và rõ ràng. Ở chiều ngược lại, một vài lĩnh vực như Thanh tra Chính phủ (51.92%) hoặc Bộ Công an (54.41%) có độ chính xác thấp hơn tương

Bảng 4.2: Accuracy của các mô hình khi không truyền kiến thức liên quan (Đơn vị: %)

Mã nguồn đóng/truy cập qua API

Mô hình	Accuracy
GPT-4o-mini	**51.72**
Gemini-2.0-flash	51.29
DeepSeek-V3-0324	50.26
Claude-3.5-Haiku	43.58

Mã nguồn mở chưa được tinh chỉnh trên tiếng Việt

Mô hình	Accuracy
LLaMA-3	53.07
Mistral-v0.3	48.24
Qwen-2.5	51.91
Vicuna-v1.5	50.59
WizardLM-2	**57.61**

Mã nguồn mở đã được tinh chỉnh trên tiếng Việt

Mô hình	Accuracy
Vistral	50.91
Qwen-Viet	**53.81**

đối. Dù vẫn cao hơn mức ngẫu nhiên, sự khác biệt này cho thấy mô hình có thể gặp khó khăn trong việc xử lý các lĩnh vực có văn bản hành chính phức tạp hoặc chứa nhiều ngoại lệ.

Bảng 4.4 trình bày độ chính xác của mô hình WizardLM-2 khi đánh giá theo bốn phân loại ảo giác. Khác với kết quả trung bình của nhiều mô hình trước đó vốn dao động quanh mức 50%, WizardLM-2 thể hiện sự phân hóa rõ ràng giữa các nhóm. Cụ thể, nhóm P-III (câu trả lời quá chung chung hoặc quá chi tiết) đạt độ chính xác cao nhất, lên đến 65.89%. Điều này cho thấy mô hình đặc biệt nhạy bén trong việc phát hiện các câu trả lời không tương xứng về mức độ chi tiết so với câu hỏi. Ngược lại, nhóm P-IV (suy luận sai từ tri thức) có độ chính xác thấp nhất, chỉ đạt 54.15%, cho thấy đây là loại ảo giác khó xử lý nhất đối với mô hình. Hai nhóm còn lại, P-I (hiểu sai ngữ cảnh và mục đích) và P-II (mâu thuẫn với

Bảng 4.3: Accuracy của mô hình WizardLM-2 khi không truyền kiến thức liên quan trên các bộ/ngành khác nhau (Đơn vị: %)

Bộ/ngành	**Accuracy**
Thanh tra Chính phủ	51.92
Bộ Khoa học và Công nghệ	57.43
Bộ Y tế	58.56
Bộ Quốc phòng	62.58
Bộ Nông nghiệp và Môi trường	59.28
Bộ Tài nguyên và Môi trường	56.45
Bộ Công thương	54.76
Bộ Tư pháp	58.94
Bộ Giao thông vận tải	55.44
Bộ Ngoại giao	57.19
Bộ Công an	54.41
Bộ Tài chính	56.16
Bộ Nội vụ	57.45
Bộ Lao động - Thương binh và Xã hội	**64.44**

tri thức), có kết quả tương đối sát nhau, lần lượt là 55.80% và 54.48%. Nhìn chung, WizardLM-2 cho thấy khả năng phân biệt tốt hơn với các dạng ảo giác hình thức (như P-III), trong khi vẫn gặp thách thức với các dạng yêu cầu suy luận sâu hoặc kiến thức nền tảng.

Bảng 4.4: Accuracy của mô hình WizardLM-2 khi không truyền kiến thức liên quan trên các phân loại ảo giác (pattern) (Đơn vị: %)

Pattern	**Mô tả**	**Accuracy**
P-I	Hiểu sai ngữ cảnh và mục đích	55.80
P-II	Mâu thuẫn giữa câu trả lời và tri thức	54.48
P-III	Quá chung chung hoặc quá chi tiết	**65.89**
P-IV	Suy luận sai từ tri thức	54.15

Ngoài ra, các thống kê bổ sung cho những mô hình khác trên phương diện bộ/ngành và phân loại ảo giác (pattern) được trình bày ở phần phụ lục D.

Bảng 4.5 so sánh độ chính xác của ba mô hình tiêu biểu trong hai thiết

lập: không truyền và có truyền kiến thức liên quan. Kết quả cho thấy việc truyền toàn bộ kiến thức nền vào prompt chưa mang lại cải thiện rõ rệt về độ chính xác, thậm chí trong một số trường hợp còn gây suy giảm. Mô hình WizardLM-2 có độ chính xác cao nhất trong thiết lập không truyền kiến thức (57.61%), nhưng khi bổ sung tri thức, chỉ số này giảm còn 47.50%. Tương tự, GPT-4o-mini giảm nhẹ từ 51.72% xuống 50.00%. Riêng Qwen-Viet giữ được độ ổn định tương đối, với mức 53.81% và 50.15% lần lượt trước và sau khi truyền tri thức. Những kết quả này cho thấy các mô hình có xu hướng thay đổi chiến lược đánh giá khi được cung cấp thêm tri thức, nhưng không nhất thiết dẫn đến hiệu quả cao hơn. Việc truyền kiến thức một cách trực tiếp vào prompt có thể chưa tối ưu hoặc làm nhiễu quá trình ra quyết định. Điều này gợi ý rằng cần tiếp tục nghiên cứu các phương pháp nâng cao hơn, chẳng hạn như tinh chỉnh mô hình (fine-tuning) hoặc tích hợp hệ thống truy xuất tri thức (RAG), nhằm cải thiện khả năng hiểu ngữ cảnh và phát hiện thông tin ảo giác một cách chính xác hơn.

Ngoài ra, thống kê bổ sung đầy đủ cho các mô hình mã nguồn mở khi truyền kiến thức liên quan cùng với bảng so sánh với số mẫu dương bằng nhau bằng 1000 cũng được trình bày ở phần phụ lục D.

Bảng 4.5: Accuracy khi không truyền và khi truyền kiến thức liên quan trên những mô hình tiêu biểu (Đơn vị: %)

Mô hình	**Không truyền kiến thức**		**Có truyền kiến thức**	
	Số mẫu dương (âm)*	**Accuracy**	**Số mẫu dương (âm)***	**Accuracy**
GPT-4o-mini	3717	51.72	1000	50.00
WizardLM-2	3717	**57.61**	1000	47.50
Qwen-Viet	3717	53.81	1000	**50.15**

*: Số lượng mẫu dương = Số lượng mẫu âm

Chương 5

Kết luận

5.1 Kết luận

Nhằm góp phần giải quyết vấn đề ảo giác trong mô hình ngôn ngữ lớn (LLM) khi ứng dụng vào ngữ cảnh dịch vụ công, luận văn đã đề xuất và xây dựng một **bộ dữ liệu chuyên biệt tập trung** vào hiện tượng ảo giác trong lĩnh vực này. Quy trình xây dựng bao gồm các bước: thu thập dữ liệu, tiền xử lý, sinh phản hồi có kiểm soát và đánh giá thủ công, nhằm đảm bảo chất lượng nhất quán của cả dữ liệu đầu vào lẫn đầu ra.

Tập dữ liệu thu được không chỉ giúp phát hiện các phản hồi ảo giác mà còn đóng vai trò là **công cụ đánh giá** hiệu quả đối với nhiều mô hình LLM khác nhau. Kết quả thực nghiệm cho thấy, **khả năng phát hiện ảo giác của mô hình còn hạn chế khi thiếu tri thức chuyên ngành**. Do đó, bộ dữ liệu được xây dựng với cả thông tin chính xác lẫn phản hồi sai lệch sẽ là nền tảng quan trọng trong việc huấn luyện và cải thiện hiệu năng của LLM, góp phần nâng cao độ chính xác và độ tin cậy khi áp dụng vào ngữ cảnh hành chính công.

5.2 Hướng phát triển

Mặc dù nghiên cứu đã xây dựng được một quy trình tạo dữ liệu tương đối hoàn chỉnh và đóng góp một tập dữ liệu có giá trị, vẫn còn một số hạn chế cần khắc phục trong các giai đoạn tiếp theo. Hiện tại, quy trình chủ yếu vận hành tự động với sự can thiệp thủ công giới hạn ở bước kiểm định đầu vào và đánh giá đầu ra. Trong bối cảnh các thủ tục hành chính thường xuyên thay đổi theo quy định pháp luật, việc sử dụng một bộ dữ liệu tĩnh có thể khiến mô hình đánh giá không phản ánh đúng thực tế.

Tuy nhiên, lợi thế của quy trình này nằm ở tính mở rộng và khả năng

tự động hóa. Trong tương lai, việc tích hợp hệ thống thu thập và cập nhật dữ liệu định kỳ từ các nguồn chính thống, chẳng hạn như Cổng Dịch vụ công Quốc gia, sẽ giúp duy trì độ chính xác và kịp thời của tập dữ liệu, đồng thời phản ánh các thay đổi chính sách một cách đầy đủ.

Bên cạnh đó, các mô hình ngôn ngữ được sử dụng hiện tại chưa được tinh chỉnh trên tập dữ liệu chuyên biệt về dịch vụ công, dẫn đến khả năng hiểu ngữ cảnh, cấu trúc câu hỏi và thông tin hành chính còn hạn chế. Điều này có thể ảnh hưởng đến chất lượng đánh giá ảo giác, đặc biệt trong việc nhận diện các nội dung sai lệch hoặc thiếu căn cứ.

Để nâng cao chất lượng nghiên cứu trong giai đoạn tiếp theo, đề tài đề xuất hai hướng phát triển chính:

- Phát triển hệ thống cập nhật dữ liệu động, đảm bảo tính thời sự và phù hợp với các thay đổi pháp lý
- Tinh chỉnh mô hình ngôn ngữ trên chính tập dữ liệu chuyên biệt, nhằm tăng cường khả năng hiểu ngữ cảnh hành chính công và cải thiện hiệu quả nhận diện ảo giác.

Những hướng tiếp cận này không chỉ giúp tăng cường độ chính xác trong phát hiện thông tin sai lệch mà còn nâng cao khả năng ứng dụng thực tiễn của LLM trong môi trường dịch vụ công, nơi yêu cầu tính chính xác và nhất quán của thông tin là yếu tố cốt lõi.

Danh mục công trình của tác giả

Tài liệu tham khảo

Tiếng Việt

[1] Dichvucong.me. *Trợ lý ảo dịch vụ công Việt Nam*. https://dichvucong.me. 2024.

[2] mradermacher. *Qwen2.5-7B-Instruct-Viet-SFT*. https://huggingface.co/mradermacher/Qwen2.5-7B-Instruct-Viet-SFT-GGUF. 2024.

[3] UoNLP. *Vistral-7B-Chat*. https://huggingface.co/uonlp/Vistral-7B-Chat-gguf. 2024.

Tiếng Anh

[4] DeepSeek AI. *DeepSeek-v3-0324*. https://huggingface.co/deepseek-ai/deepseek-llm-7b-chat. 2024.

[5] Meta AI. *Llama-3-7B-Q4_K_M*. https://huggingface.co/christopherBR/Llama-3-7B-Q4_K_M. 2024.

[6] Mistral AI. *Mistral-7B-Instruct-v0.3*. https://huggingface.co/lmstudio-community/Mistral-7B-Instruct-v0.3-GGUF. 2024.

[7] Anthropic. *Claude-3.5-Haiku*. https://www.anthropic.com/news/claude-3-5-haiku. 2024.

[8] Tom B. Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

[9] Shiqi Chen et al. "FELM: Benchmarking Factuality Evaluation of Large Language Models". In: *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*. 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/8b8a7960d343e023a6a0afe37eee6022-Abstract-Datasets_and_Benchmarks.html.

[10] Alibaba Cloud. *Qwen2.5-7B-Instruct-1M*. https://huggingface.co/lmstudio-community/Qwen2.5-7B-Instruct-1M-GGUF. 2024.

[11] Google DeepMind. *Gemini-2.0-flash*. https://deepmind.google/technologies/gemini/. 2024.

[12] Nouha Dziri et al. "Evaluating attribution in dialogue systems: The BEGIN benchmark". In: *Transactions of the Association for Computational Linguistics (TACL)* 10 (2022), pp. 1066–1083.

[13] Ryan T. McDonald Joshua Maynez Shashi Narayan Bernd Bohnet. "On Faithfulness and Factuality in Abstractive Summarization". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Association for Computational Linguistics, 2020, pp. 1906–1919. DOI: 10.18653/V1/2020.ACL-MAIN.173. URL: https://doi.org/10.18653/v1/2020.acl-main.173.

[14] Wayne Xin Zhao Jian-Yun Nie Ji-Rong Wen Junyi Li Xiaoxue Cheng. "HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models". In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*. Association for Computational Linguistics, 2023, pp. 6449–6464. URL: https://doi.org/10.18653/v1/2023.emnlp-main.397.

[15] LMSys. *Vicuna-7B-v1.5*. https://huggingface.co/TheBloke/vicuna-7B-v1.5-GGUF. 2023.

[16] OpenAI. *GPT-4o-mini*. https://openai.com/index/gpt-4o. 2024.

[17] Mark J. F. Gales Potsawee Manakul Adian Liusie. “SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models”. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6 10, 2023*. Association for Computational Linguistics, 2023, pp. 9004–9017. DOI: 10.18653/V1/2023.EMNLP-MAIN.557. URL: https://doi.org/10.18653/v1/2023.emnlp-main.557.

[18] Hannah Rashkin et al. “Measuring Attribution in Natural Language Generation Models”. In: *Comput. Linguistics (COLING)* 49.4 (2023), pp. 777–840. URL: https://doi.org/10.1162/coli_a_00486.

[19] Microsoft Research. *WizardLM-2-7B*. https://huggingface.co/lmstudio-community/WizardLM-2-7B-GGUF. 2024.

Chương A

Mô tả các trường dữ liệu

Bảng A.1: Mô tả cấu trúc bộ dữ liệu câu hỏi – câu trả lời

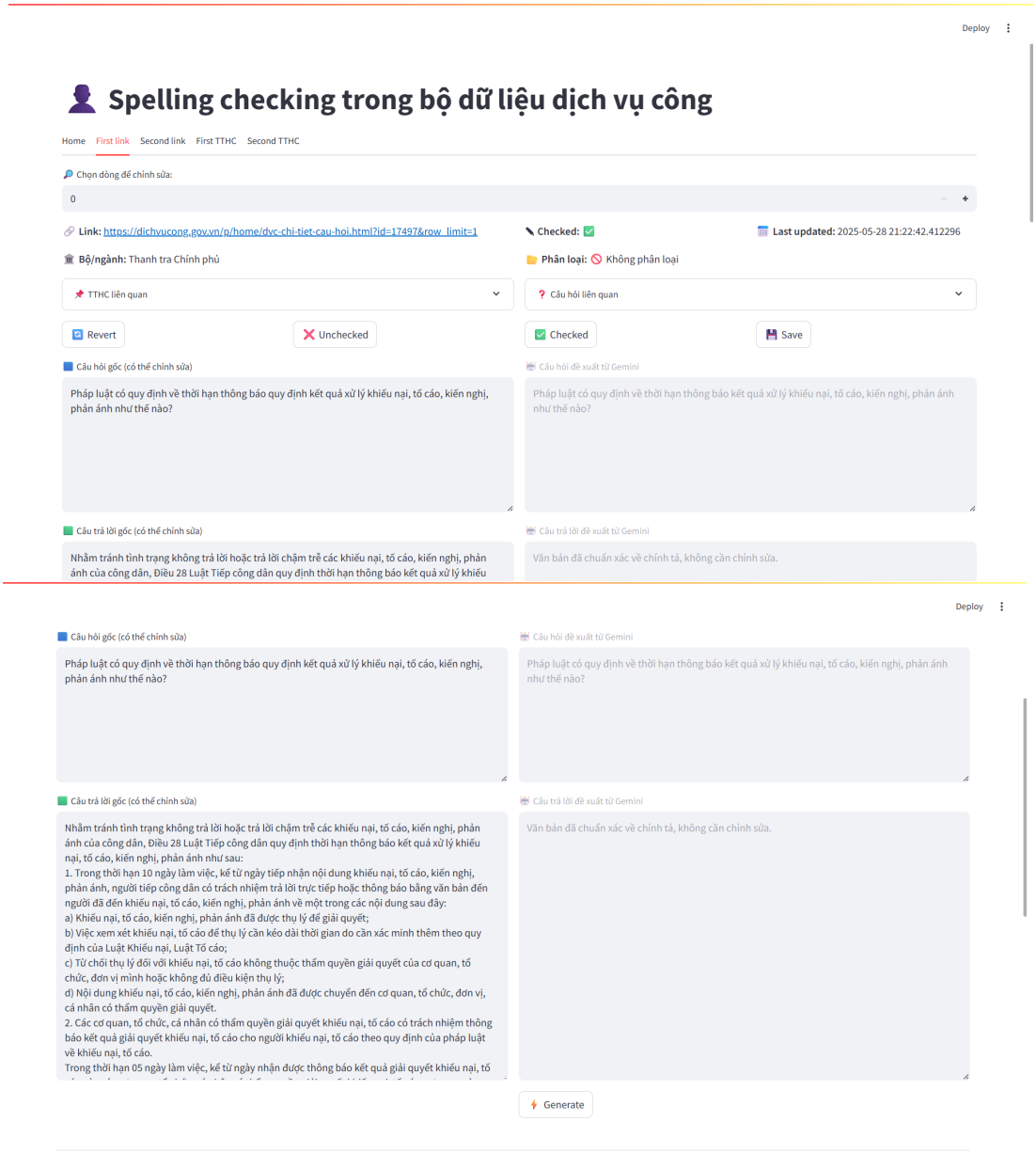
Cột	Mô tả
`link`	Đường dẫn đến trang dịch vụ công chứa thông tin gốc của thủ tục.
`phanLoai`	Nhóm phân loại nội dung câu hỏi (có thể là chủ đề hành chính).
`boNganh`	Tên bộ/ngành quản lý thủ tục hành chính liên quan.
`cauHoi`	Câu hỏi tự nhiên được trích xuất hoặc sinh ra từ nội dung thủ tục.
`cauTraLoi`	Câu trả lời đúng, được xây dựng từ dữ liệu chuẩn hóa từ dịch vụ công.
`TTHCLienQuan`	Mã thủ tục hành chính liên quan (có thể ở dạng văn bản hoặc danh sách).
`cauHoiLienQuan`	Câu hỏi liên quan khác, hỗ trợ đánh giá thông tin.
`cauTraLoiAoGiac`	Câu trả lời chứa thông tin ảo giác, được sinh bởi mô hình ngôn ngữ.
`pattern`	Nhãn ảo giác từ 0 đến 3, tương ứng với bốn loại phân loại lỗi (P-I đến P-IV).

Bảng A.2: Mô tả bộ dữ liệu kiến thức liên quan (TTHC)

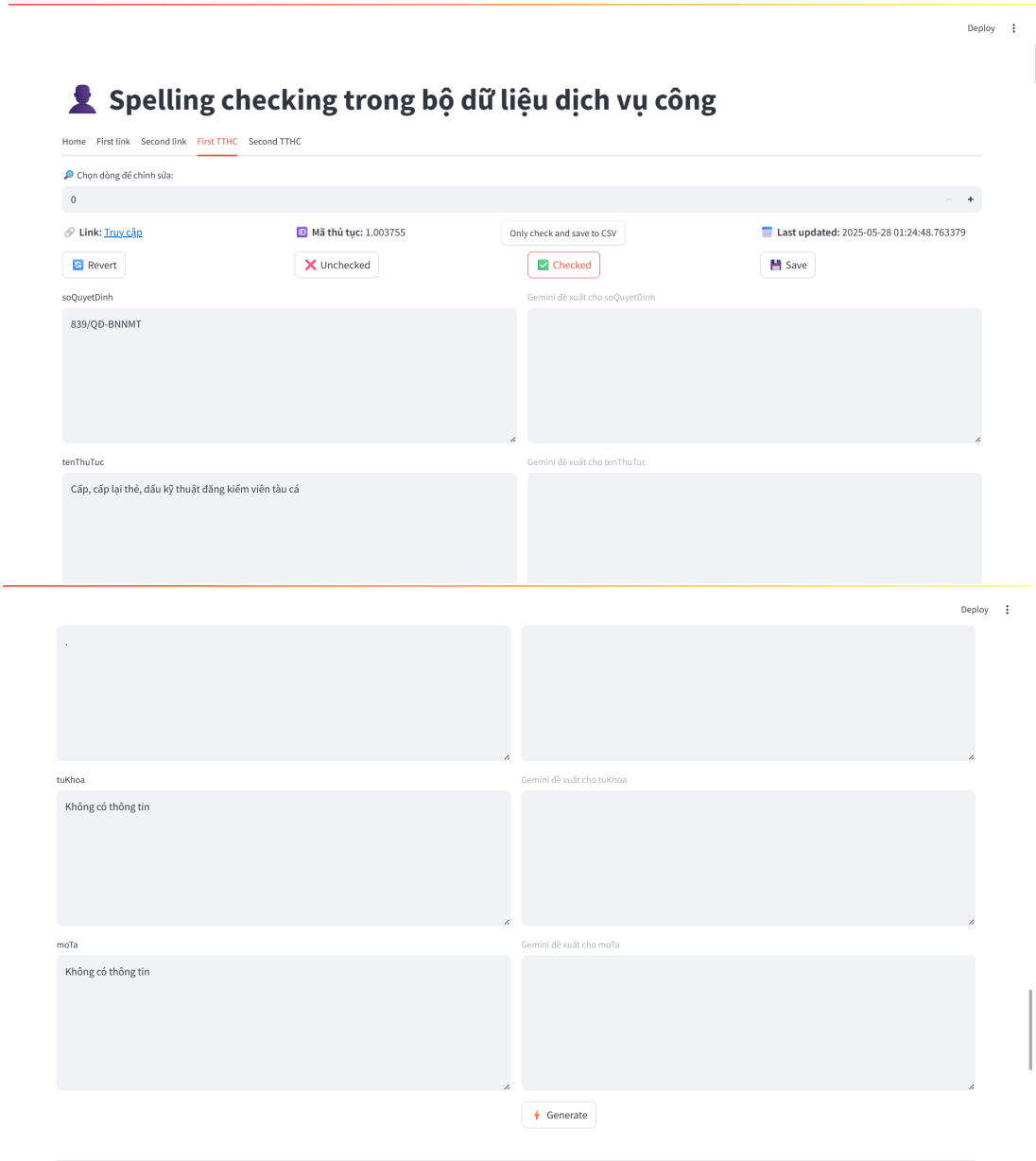
Cột	Mô tả
link	Đường dẫn đến trang gốc trên Cổng Dịch vụ công chứa thông tin về thủ tục.
maThuTuc	Mã định danh duy nhất của thủ tục hành chính.
soQuyetDinh	Số quyết định ban hành hoặc điều chỉnh thủ tục.
tenThuTuc	Tên thủ tục hành chính.
capThucHien	Cấp hành chính chịu trách nhiệm thực hiện thủ tục (Trung ương, tỉnh, huyện, xã...).
loaiThuTuc	Loại thủ tục hành chính (ví dụ: cấp mới, điều chỉnh, gia hạn...).
linhVuc	Lĩnh vực chuyên môn mà thủ tục thuộc về (như giao thông, y tế, giáo dục...).
trinhTuThucHien	Mô tả tuần tự các bước cần thực hiện để hoàn tất thủ tục.
cachThucThucHien	Hình thức nộp hồ sơ và xử lý thủ tục (trực tiếp, trực tuyến...).
thanhPhanHoSo	Danh sách các giấy tờ, biểu mẫu cần nộp trong hồ sơ.
doiTuongThucHien	Đối tượng được quyền thực hiện thủ tục (cá nhân, tổ chức...).
coQuanThucHien	Cơ quan trực tiếp tiếp nhận và xử lý hồ sơ.
coQuanCoThamQuyen	Cơ quan có thẩm quyền ra quyết định cuối cùng trong thủ tục.
diaChiTiepNhanHoSo	Địa chỉ nơi tiếp nhận hồ sơ.
coQuanDuocUyQuyen	Cơ quan được ủy quyền xử lý thủ tục (nếu có).
coQuanPhoiHop	Cơ quan phối hợp cùng xử lý trong quá trình thực hiện thủ tục.
ketQuaThucHien	Kết quả đầu ra của thủ tục (ví dụ: giấy phép, chứng nhận...).
canCuPhapLy	Danh sách văn bản pháp lý liên quan làm căn cứ thực hiện.
yeuCauDieuKienThucHien	Các yêu cầu, điều kiện bắt buộc để được thực hiện thủ tục.
tuKhoa	Từ khóa mô tả nội dung chính của thủ tục (phục vụ tìm kiếm, phân loại).
moTa	Mô tả tổng quan về thủ tục, tóm tắt mục đích và phạm vi áp dụng.

Chương B

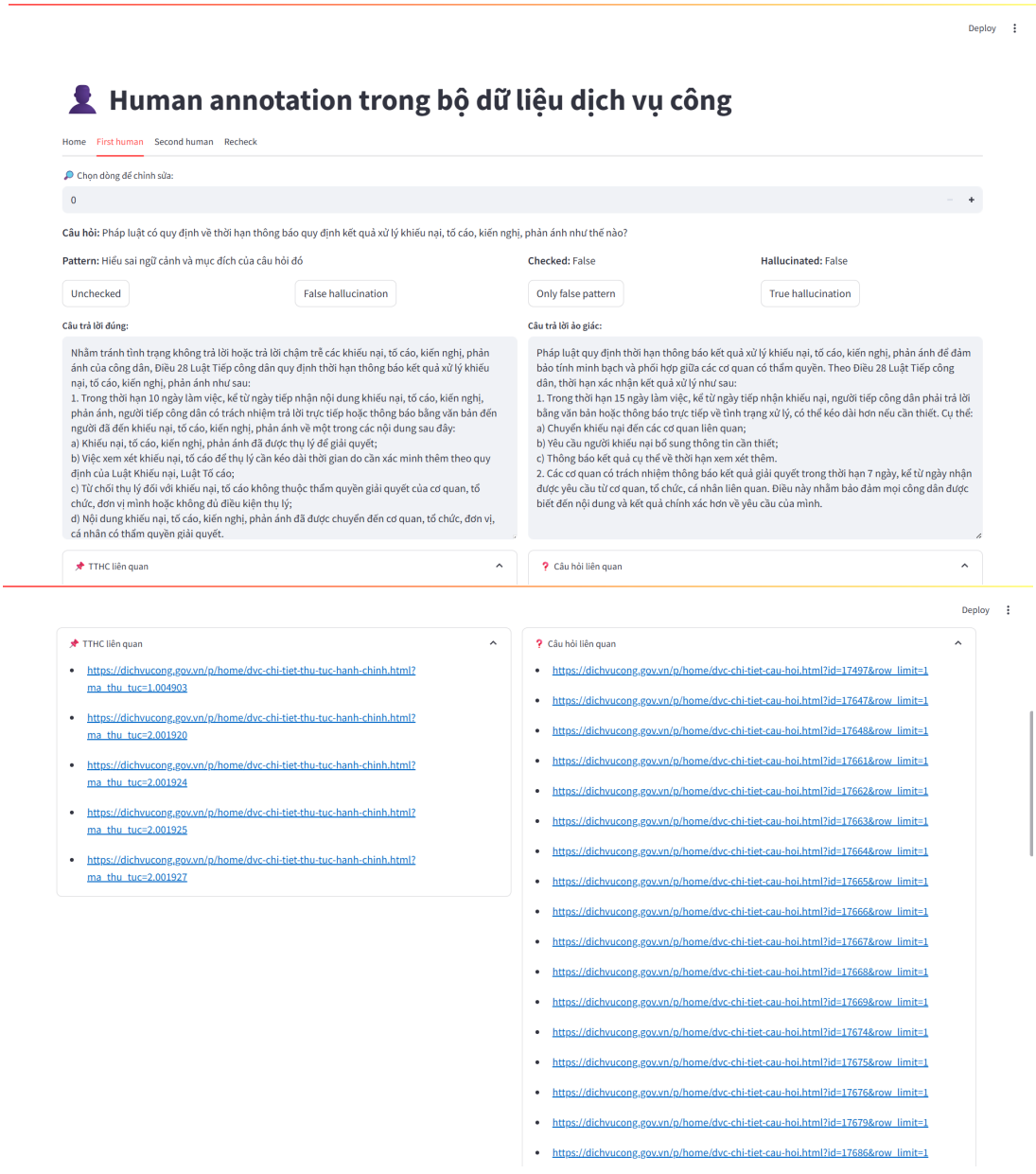
Giao diện thủ công



Hình B.1: Giao diện kiểm tra chính tả thủ công cho bộ dữ liệu câu hỏi-câu trả lời



Hình B.2: Giao diện kiểm tra chính tả thủ công cho bộ dữ liệu kiến thức liên quan (TTHC)



Hình B.3: Giao diện chú thích ảo giác cho bộ dữ liệu dịch vụ công

Chương C

Mẫu truy vấn

Bảng C.1: Truy vấn đánh giá (sử dụng kiến thức liên quan)

Truy vấn hệ thống:
Bạn sẽ đóng vai trò là một người đánh giá câu trả lời (answer judge). Với một câu hỏi, câu trả lời, và kiến thức liên quan, mục tiêu của bạn là xác định xem câu trả lời được cung cấp có chứa thông tin không đúng sự thật hoặc thông tin ảo giác (hallucinated information) hay không. *{pattern}* Bạn nên cố gắng hết sức để xác định xem câu trả lời có chứa thông tin không đúng sự thật hoặc thông tin ảo giác hay không. Câu trả lời bạn đưa ra bắt buộc CHỈ là "Có" hoặc "Không", và không giải thích gì thêm. Trả lời "Có" nếu câu trả lời chứa thông tin ảo giác, trả lời "Không" nếu câu trả lời không chứa thông tin ảo giác.
Truy vấn người dùng (học ngữ cảnh không có ví dụ):
#Kiến thức liên quan#: *{knowledge}* #Câu hỏi#: *{question}* #Câu trả lời#: *{answer}* #Đánh giá của bạn#:

Chương D

Thống kê bổ sung

Bảng D.1: Bảng thống kê đánh giá các mô hình khi không truyền kiến thức liên quan sử dụng các độ đo khác nhau (Đơn vị: %)

Mô hình	**Accuracy**	**Precision**	**Recall**	**F1-score**
GPT-4o-mini	51.72	88.40	50.99	64.68
Gemini-2.0-flash	51.29	69.65	50.94	58.85
DeepSeek-V3-0324	50.26	85.88	50.15	63.32
Claude-3.5-Haiku	43.58	25.69	40.01	31.29
Llama-3	53.07	76.11	52.10	61.86
Mistral-v0.3	48.24	44.85	48.11	46.42
Qwen-2.5	51.91	89.05	51.10	64.93
Vicuna-v1.5	50.59	**99.68**	50.30	**66.86**
WizardLM-2	**57.61**	44.79	**60.24**	51.38
Vistral	50.91	6.67	57.94	11.97
Qwen-Viet	53.81	75.76	52.65	62.12

Bảng D.2: Accuracy trung bình trên các mô hình theo từng bộ/ngành (Đơn vị: %)

Bộ/ngành	**Accuracy**
Thanh tra Chính phủ	50.38
Bộ Tư pháp	50.42
Bộ Khoa học và CN	51.22
Bộ Giao thông vận tải	51.66
Bộ Y tế	50.71
Bộ Ngoại giao	51.57
Bộ Quốc phòng	52.07
Bộ Công an	50.70
Bộ NN & MT	51.56
Bộ Tài chính	50.09
Bộ TN & MT	50.81
Bộ Nội vụ	51.22
Bộ Công thương	51.08
Bộ Lao động – Thương binh & XH	**52.12**

Bảng D.3: Accuracy trung bình trên các mô hình theo từng phân loại ảo giác (Đơn vị: %)

Pattern	**Accuracy**
P-I	51.59
P-II	**52.73**
P-III	51.97
P-IV	48.48

Bảng D.4: So sánh hiệu suất khi không truyền và khi truyền kiến thức liên quan trên những mô hình mã nguồn đóng và hai mô hình mã nguồn mở tiêu biểu (Đơn vị: %)

Mô hình	**Có tri thức**	**Số mẫu dương (âm)**	**Accuracy**	**Recall**	**Precision**	**F1-score**
GPT-4o-mini	Không	3717	51.72	88.40	50.99	64.68
	Có	1000	50.00	**99.40**	50.00	**66.53**
Gemini-2.0-flash	Không	3717	51.29	69.65	50.94	58.85
	Có	1000	49.85	92.40	49.92	64.82
DeepSeek-V3-0324	Không	3717	50.26	85.88	50.15	63.32
	Có	1000	49.90	98.10	49.95	66.19
Claude-3.5-Haiku	Không	3717	43.58	25.69	40.01	31.29
	Có	1000	45.55	33.50	44.14	38.09
WizardLM-2	Không	3717	**57.61**	44.79	**60.24**	51.38
	Có	1000	47.50	28.30	45.94	35.02
Qwen-Viet	Không	3717	53.81	75.76	52.65	62.12
	Có	1000	50.15	96.70	50.08	65.98

Bảng D.5: Accuracy khi không truyền và khi truyền kiến thức liên quan (số mẫu ngang nhau) trên những mô hình tiêu biểu (Đơn vị: %)

Mô hình	**Không truyền kiến thức**		**Có truyền kiến thức**	
	Số mẫu dương (âm)*	**Accuracy**	**Số mẫu dương (âm)***	**Accuracy**
GPT-4o-mini	1000	51.80	1000	50.00
WizardLM-2	1000	**58.60**	1000	47.50
Qwen-Viet	1000	53.80	1000	**50.15**

*: Số lượng mẫu dương = Số lượng mẫu âm