HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



BÙI KHẮC NGỌC

PHÁT TRIỂN CÔNG CỤ ỨNG DỤNG AI TRONG VIỆC CHUYỂN ĐỔI NGÔN NGỮ TỰ NHIÊN THÀNH CÂU TRUY VẤN SQL NHẰM HỖ TRỢ TRA CỨU NHANH THÔNG TIN

Chuyên ngành: Hệ thống thông tin

Mã số: 8.48.01.04

TÓM TẮT ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS.TS. NGÔ QUỐC DŨNG

HÀ NỘI - 2025

Đề án tốt nghiệp được hoàn thành tại:

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: ……

(Ghi rõ học hàm, học vị)

Phản biện 1: …….

Phản biện 2: ……

Đề án tốt nghiệp sẽ được bảo vệ trước Hội đồng chấm đề án tốt nghiệp thạc sĩ tại Học viện Công nghệ Bưu chính Viễn thông

Vào lúc: ..... giờ ..... ngày ..... tháng .... năm .....

Có thể tìm hiểu đề án tốt nghiệp tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông.

MỞ ĐẦU

1. Lựa chọn đề tài

Trong bối cảnh chuyển đổi số quốc gia theo Quyết định số 749/QĐ-TTg, ngành giáo dục đại học Việt Nam đang đẩy mạnh triển khai mô hình đại học số. Việc xây dựng và vận hành hệ thống quản lý dữ liệu tập trung đặt ra những thách thức về khả năng khai thác và truy xuất thông tin hiệu quả. Thực tế cho thấy người dùng không có chuyên môn kỹ thuật gặp nhiều khó khăn khi cần truy vấn dữ liệu từ hệ thống, do thiếu kiến thức về SQL và cấu trúc cơ sở dữ liệu.

Sự phát triển của trí tuệ nhân tạo, đặc biệt là các mô hình ngôn ngữ lớn (Large Language Models - LLM), mở ra khả năng chuyển đổi tự động từ ngôn ngữ tự nhiên sang câu truy vấn SQL. Tuy nhiên, hầu hết các nghiên cứu hiện tại tập trung vào tiếng Anh, trong khi tiếng Việt với những đặc thù riêng về ngữ pháp, từ vựng và cách diễn đạt chưa được quan tâm đúng mức. Đây là khoảng trống nghiên cứu cần được lấp đầy.

Xuất phát từ thực tiễn và nhu cầu cấp thiết đó, đề tài "Xây dựng công cụ chuyển đổi ngôn ngữ tiếng Việt thành câu truy vấn SQL cho Đại học số" được lựa chọn nhằm nghiên cứu và phát triển giải pháp giúp người dùng có thể truy xuất thông tin từ cơ sở dữ liệu thông qua câu hỏi bằng tiếng Việt tự nhiên, không cần kiến thức về SQL.

2. Mục đích nghiên cứu

Đề tài hướng đến các mục tiêu cụ thể sau:

Thứ nhất, thiết kế và triển khai một hệ thống Text-to-SQL hoàn chỉnh dựa trên nguyên lý Retrieval-Augmented Generation (RAG) và Semantic Layer, có khả năng xử lý câu hỏi tiếng Việt với độ chính xác cao và thời gian phản hồi tối ưu.

Thứ hai, xây dựng các tài nguyên nền tảng bao gồm chuẩn hóa schema cơ sở dữ liệu, phát triển bộ từ điển dữ liệu chuyên biệt cho lĩnh vực giáo dục đại học, và tạo bộ câu hỏi mẫu phục vụ huấn luyện và đánh giá.

Thứ ba, tích hợp mô hình ngôn ngữ lớn thông qua các kỹ thuật Prompt Engineering và cơ chế truy xuất tăng cường để thực hiện chuyển đổi chính xác từ ngôn ngữ tiếng Việt sang câu truy vấn SQL.

Thứ tư, đảm bảo tính thực tiễn và bảo mật thông qua việc triển khai các cơ chế an toàn, chỉ cho phép thực hiện truy vấn đọc dữ liệu, cung cấp giao diện người dùng trực quan.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu: Đề tài tập trung vào việc xây dựng công cụ hướng đến người dùng là ban lãnh đạo trường đại học, ban giám đốc học viện, cùng với các chuyên viên, giảng viên của các đơn vị trực thuộc cơ sở giáo dục đại học. Công cụ ứng dụng LLM để hiểu và xử lý câu hỏi tiếng Việt, các kỹ thuật ánh xạ câu hỏi với cơ sở dữ liệu.

Phạm vi nghiên cứu: Đề tài giới hạn trên bốn phương diện chính: (1) Tập trung vào hệ quản trị cơ sở dữ liệu quan hệ PostgreSQL; (2) Hệ thống chỉ hỗ trợ các thao tác đọc dữ liệu (SELECT), không thực hiện ghi, sửa, xóa dữ liệu; (3) Nội dung ứng dụng tập trung vào dữ liệu đặc thù của quản lý đào tạo đại học bao gồm thông tin sinh viên, ngành học, chương trình đào tạo, học phần, điểm; (4) Xử lý các câu truy vấn được diễn đạt bằng tiếng Việt.

4. Phương pháp nghiên cứu

Nghiên cứu được triển khai dựa trên ba trụ cột: nền tảng lý thuyết, mô hình hóa thực nghiệm và kiểm chứng khả năng triển khai thực tiễn. Các hoạt động nghiên cứu tuân theo chuỗi logic từ thiết lập khung lý thuyết, qua giai đoạn kiến tạo hệ thống, đến khâu thẩm định và xác minh hiệu quả.

Về phương pháp nghiên cứu lý thuyết, đề tài khảo sát các công cụ Text-to-SQL hiện có, phân tích kiến trúc cùng ưu nhược điểm của từng hướng tiếp cận, nghiên cứu đặc trưng tiếng Việt trong bối cảnh chuyển đổi sang SQL.

Về phương pháp thực nghiệm, đề tài tiến hành xây dựng từ điển dữ liệu, chuẩn hóa schema, phát triển và tích hợp hệ thống dựa trên nền tảng WrenAI. Quy trình kiểm thử được triển khai với bộ dữ liệu 1139 câu hỏi mẫu để đánh giá độ chính xác và hiệu năng.

5. Cấu trúc đề án

Ngoài phần Mở đầu, Kết luận và Phụ lục, nội dung chính của đề án gồm ba chương:

Chương 1: Tổng quan về bài toán - Trình bày cơ sở lý luận về cơ sở dữ liệu, đại học số, trí tuệ nhân tạo và bài toán Text-to-SQL.

Chương 2: Phân tích, đánh giá và thiết kế giải pháp - Phân tích yêu cầu hệ thống, nghiên cứu các phương pháp chuẩn hóa schema, xây dựng từ điển dữ liệu, thiết kế kiến trúc và khảo sát công cụ hiện có.

Chương 3: Xây dựng và triển khai hệ thống - Trình bày chi tiết quá trình triển khai, xây dựng tài nguyên nền tảng, phát triển các module, thử nghiệm và đánh giá kết quả.

CHƯƠNG 1: TỔNG QUAN VỀ BÀI TOÁN

1.1. Tổng quan về cơ sở dữ liệu và hệ thống quản lý

Cơ sở dữ liệu được hình thành từ tập hợp các thông tin được tổ chức có cấu trúc, lưu trữ dưới dạng số và truy cập thông qua hệ thống máy tính. Cơ sở dữ liệu đóng vai trò nền tảng trong tổ chức hiện đại, đảm bảo việc lưu trữ, quản lý và khai thác thông tin phục vụ vận hành và quá trình ra quyết định.

Hệ quản trị cơ sở dữ liệu (DBMS) đóng vai trò lớp trung gian giữa người dùng và dữ liệu, cho phép thực hiện các thao tác thêm, sửa, xóa, xem dữ liệu một cách kiểm soát và nhất quán. Trong các mô hình cơ sở dữ liệu, mô hình quan hệ được sử dụng phổ biến nhất vì dữ liệu được lưu thành các bảng liên kết với nhau, với SQL là ngôn ngữ chuẩn ISO/ANSI để làm việc.

Ngoài mô hình quan hệ, còn có mô hình NoSQL được triển khai cho các tập dữ liệu phi cấu trúc hoặc có tính động cao. Các nền tảng như Amazon Redshift, Google BigQuery được xây dựng để đáp ứng các tác vụ tính toán truy xuất với quy mô dữ liệu lớn. Tuy nhiên, cơ sở dữ liệu quan hệ vẫn được sử dụng nhiều trong quản lý dữ liệu có cấu trúc tại các tổ chức giáo dục, tài chính, y tế và hành chính công.

1.2. Tổng quan về đại học số và nhu cầu truy vấn dữ liệu

Đại học số (Digital University) là mô hình triển khai công nghệ thông tin đồng bộ và toàn diện vào mọi hoạt động quản lý và hỗ trợ học thuật trong môi trường giáo dục đại học. Theo Nghị quyết 71 về phát triển giáo dục, việc xây dựng và triển khai đại học số sẽ nâng cao chất lượng giảng dạy, cải thiện hiệu quả vận hành và năng lực điều hành trong các cơ sở đào tạo đại học.

Kiến trúc của đại học số bao gồm các phần mềm quản lý sinh viên, đào tạo, nhân sự, thư viện, tài chính. Yếu tố cốt lõi là dữ liệu số, trong đó toàn bộ thông tin sinh viên, giảng viên, chương trình đào tạo, học phần, ngành học được số hóa, lưu trữ và quản lý tập trung.

Hiện nay, quá trình khai thác dữ liệu trong đại học số chủ yếu thông qua: (i) truy cập báo cáo dựng sẵn trên giao diện web - vốn thiếu tính linh hoạt; (ii) yêu cầu bộ phận

công nghệ thông tin cung cấp khi cần truy vấn đặc thù; (iii) xuất dữ liệu thủ công dưới dạng Excel; hoặc (iv) sử dụng công cụ Business Intelligence. Người dùng là cán bộ, giảng viên gặp nhiều khó khăn trong việc truy vấn dữ liệu do thiếu kiến thức về SQL và cấu trúc cơ sở dữ liệu.

Từ thực trạng này, nhu cầu về công cụ cho phép người dùng tra cứu dữ liệu bằng tiếng Việt tự nhiên trở nên cấp thiết. Hệ thống cần có khả năng: diễn giải chính xác mục đích câu hỏi; ánh xạ câu hỏi sang mô hình dữ liệu; sinh câu truy vấn SQL chính xác và an toàn; trình bày kết quả dễ hiểu; đảm bảo tốc độ phản hồi; và cung cấp giao diện người dùng trực quan.

1.3. Tổng quan về trí tuệ nhân tạo và mô hình ngôn ngữ lớn

Trí tuệ nhân tạo (AI) là ngành khoa học máy tính nghiên cứu việc tái tạo trong máy móc những năng lực nhận thức của con người như nhận diện hình ảnh, xử lý ngôn ngữ, ra quyết định và học từ kinh nghiệm. Trong AI, có nhiều hướng tiếp cận chính: AI dựa trên luật sẵn có; học máy nơi hệ thống học từ dữ liệu; học sâu khai thác mạng nơ-ron nhiều lớp; học tăng cường nơi hệ thống học thông qua thử nghiệm trong môi trường tương tác.

Xử lý ngôn ngữ tự nhiên (NLP) là hướng nghiên cứu cho phép máy tính hiểu và xử lý ngôn ngữ con người. Đối với tiếng Việt, lĩnh vực này phải đối diện với các thách thức từ đặc tính ngôn ngữ học phức tạp: tính nhập nhằng, sự phụ thuộc cao vào ngữ cảnh, đa dạng cấu trúc diễn đạt và nhu cầu về tri thức nền.

Mô hình ngôn ngữ lớn (Large Language Models - LLM) là các mô hình học sâu được huấn luyện trên khối lượng văn bản khổng lồ, có thể hiểu và tạo ra văn bản tự nhiên với chất lượng cao. LLM có các đặc điểm: quy mô lớn; khả năng học từ ngữ cảnh để hiểu ý nghĩa mà không cần huấn luyện lại; tính đa năng trong việc thực hiện nhiều nhiệm vụ khác nhau chỉ với prompt phù hợp; few-shot learning để học từ vài ví dụ trong prompt; zero-shot learning để thực hiện nhiệm vụ mới mà không cần ví dụ.

Các mô hình ngôn ngữ lớn tiêu biểu gồm: BERT của Google với thế mạnh về hiểu ngôn ngữ; GPT của OpenAI với khả năng sinh văn bản; T5 của Google để xử lý

mọi nhiệm vụ NLP như chuyển đổi text-to-text; LLaMA của Meta là mô hình mã nguồn mở; Claude của Anthropic; PaLM của Google.

1.4. Bài toán chuyển đổi ngôn ngữ tiếng Việt thành SQL

Text-to-SQL là bài toán chuyển đổi câu hỏi bằng ngôn ngữ tự nhiên thành câu truy vấn SQL có thể thực thi trên cơ sở dữ liệu. Bài toán trải qua ba giai đoạn phát triển: giai đoạn dựa trên luật với NLIDB trong những năm 1980-1990; giai đoạn học máy với Seq2Seq và attention trong những năm 2010; giai đoạn mô hình ngôn ngữ lớn từ 2020 với GPT-4, Claude mang lại bước nhảy đột phá về chất lượng.

Các thách thức chính trong bài toán: (1) Tính phức tạp của ngôn ngữ tự nhiên với sự mơ hồ, đa nghĩa và đa dạng cách diễn đạt; (2) Khoảng cách giữa ngôn ngữ tự nhiên và SQL trong việc ánh xạ khái niệm trừu tượng; (3) Tính phức tạp của schema với hàng trăm bảng và mối quan hệ; (4) Yêu cầu về hiểu biết nghiệp vụ để sinh truy vấn chính xác.

Đặc thù của tiếng Việt trong bài toán Text-to-SQL đặt ra những thách thức riêng: tiếng Việt là ngôn ngữ đơn lập với ranh giới từ không rõ ràng đòi hỏi tách từ (word segmentation); hệ thống thanh điệu tạo ra thách thức khi người dùng nhập không dấu; cấu trúc ngữ pháp khác biệt với tiếng Anh; thuật ngữ giáo dục có nhiều biến thể và cách viết tắt đặc thù.

1.5. Kết luận chương 1

Chương 1 đã trình bày cơ sở lý luận về cơ sở dữ liệu, đại học số, trí tuệ nhân tạo và bài toán Text-to-SQL. Việc nghiên cứu cho thấy nhu cầu cấp thiết về công cụ chuyển đổi tiếng Việt thành SQL trong bối cảnh đại học số, đồng thời xác định các thách thức về ngôn ngữ và kỹ thuật cần giải quyết. Đây là nền tảng lý thuyết để triển khai phân tích và thiết kế giải pháp trong Chương 2.

CHƯƠNG 2: PHÂN TÍCH, ĐÁNH GIÁ VÀ THIẾT KẾ GIẢI PHÁP

2.1. Phân tích yêu cầu hệ thống

2.1.1. Yêu cầu chức năng

Yêu cầu về chuyển đổi và sinh câu truy vấn: Thực hiện schema linking để ánh xạ các thành phần trong câu hỏi với bảng, cột và thuộc tính trong cơ sở dữ liệu. Hệ thống phải sinh câu SQL có cú pháp hợp lệ, hỗ trợ các truy vấn phức tạp như kết nối nhiều bảng (JOIN), truy vấn lồng nhau (subquery), phép tổng hợp (aggregation), nhóm dữ liệu (GROUP BY), lọc kết quả và sắp xếp.

Yêu cầu về thực thi và trả kết quả: Đảm bảo tính an toàn với kiểm tra ngăn chặn SQL injection, chỉ cho phép thực hiện truy vấn ở chế độ READ-ONLY. Kết quả được định dạng dễ hiểu với bảng dữ liệu, tiêu đề tiếng Việt, chuẩn hóa định dạng số và ngày tháng. Hỗ trợ xuất dữ liệu và trực quan hóa biểu đồ.

2.1.2. Yêu cầu phi chức năng

Về hiệu năng: Thời gian phản hồi dưới 2-3 giây với truy vấn đơn giản và 5-7 giây với truy vấn phức tạp. Khi thời gian xử lý vượt quá 10 giây, cần cung cấp cơ chế thông báo tiến độ. Hỗ trợ xử lý đồng thời nhiều truy vấn, tối ưu mức sử dụng tài nguyên (CPU, RAM, băng thông, số lượng API call đến LLM).

Về độ chính xác: Đạt tối thiểu 80% câu truy vấn SQL có cú pháp hợp lệ và khả thi; với truy vấn phức tạp ngưỡng chấp nhận 70%. Mục tiêu đạt 75-80% Execution Accuracy. Trong trường hợp không chắc chắn, hệ thống cần có cơ chế hỏi lại người dùng hoặc cung cấp các cách diễn giải thay thế.

2.2. Nghiên cứu và đánh giá các phương pháp chuẩn hóa schema

Schema cơ sở dữ liệu là mô tả hình thức về cấu trúc dữ liệu, bao gồm các bảng, thuộc tính, kiểu dữ liệu, khóa chính, khóa ngoại và các ràng buộc toàn vẹn. Trong bài toán Text-to-SQL, schema đóng vai trò nền tảng định hướng mô hình trong việc hiểu cấu trúc dữ liệu và xác định mối liên hệ giữa các thực thể.

Các hoạt động chuẩn hóa bao gồm: (1) Truy vấn metadata để thu thập thông tin cấu trúc từ DBMS: danh sách bảng, cột, kiểu dữ liệu, khóa và ràng buộc; (2) Làm giàu thông tin ngữ nghĩa bằng mô tả chi tiết tiếng Việt cho từng thành phần; (3) Chuẩn hóa tên gọi từ dạng kỹ thuật sang dạng dễ hiểu; (4) Xác định mối quan hệ ngữ nghĩa giữa các bảng; (5) Bổ sung ví dụ giá trị mẫu; (6) Định nghĩa business rules.

Đề tài đề xuất phương pháp kết hợp: JSON Schema làm nền tảng cấu trúc để lưu trữ metadata cơ bản; Natural Language Descriptions bằng tiếng Việt để làm giàu ngữ nghĩa với mô tả ý nghĩa nghiệp vụ, quy tắc liên quan; Linearization được tối ưu cho LLM với ưu tiên các bảng có khả năng liên quan cao, sắp xếp theo mức độ quan trọng, format ngắn gọn để tiết kiệm token.

2.3. Nghiên cứu xây dựng bộ từ điển dữ liệu

Từ điển dữ liệu (Data Dictionary) đóng vai trò quan trọng trong việc kết nối ngôn ngữ tiếng Việt với cấu trúc cơ sở dữ liệu. Trong bối cảnh Text-to-SQL cho đại học số, từ điển không chỉ là danh sách thuật ngữ mà còn là cầu nối giữa cách diễn đạt của người Việt với schema database thường được đặt tên theo quy tắc lập trình.

Vai trò của từ điển dữ liệu trong Text-to-SQL: (1) Hỗ trợ schema linking để nhận diện bảng/cột khi người dùng đề cập; (2) Xử lý đa ngôn ngữ và đa cách diễn đạt với các biến thể của cùng một khái niệm; (3) Cung cấp ngữ cảnh nghiệp vụ với định nghĩa, ví dụ và quy tắc; (4) Hỗ trợ xử lý câu hỏi mơ hồ với các khả năng diễn giải; (5) Tăng độ chính xác của LLM bằng cách cung cấp định nghĩa rõ ràng; (6) Hỗ trợ mở rộng và bảo trì hệ thống.

Quy trình xây dựng từ điển gồm ba giai đoạn: Giai đoạn thu thập từ tài liệu chính thức (quy chế, quy định Bộ GD&ĐT), hệ thống thông tin hiện có (phân tích schema database), người dùng thực tế (phỏng vấn cán bộ, giảng viên), câu hỏi mẫu từ các kênh tương tác. Giai đoạn phân loại và chuẩn hóa theo nhóm thực thể, xác định thuật ngữ chuẩn, viết định nghĩa. Giai đoạn thiết lập ánh xạ từ thuật ngữ nghiệp vụ đến các thành phần cụ thể trong schema.

2.4. Nghiên cứu thiết kế Semantic Layer cho hệ thống

Semantic Layer là một lớp trừu tượng nằm giữa câu hỏi ngôn ngữ tự nhiên và cấu trúc cơ sở dữ liệu vật lý. Theo nghiên cứu của AtScale và Denodo, Semantic Layer có thể cải thiện độ chính xác từ 20% lên 92.5% khi được triển khai đúng cách. Semantic Layer đóng vai trò như kho lưu trữ tập trung cho logic nghiệp vụ, các chỉ số KPI và mối quan hệ giữa các bảng.

Vai trò của Semantic Layer bao gồm: (1) Mã hóa logic nghiệp vụ và các quy tắc vào định dạng có cấu trúc; (2) Thống nhất thuật ngữ nghiệp vụ, ánh xạ từ thuật ngữ kinh doanh sang tên kỹ thuật trong cơ sở dữ liệu; (3) Quản lý mối quan hệ rõ ràng giữa các thực thể, giúp mô hình hiểu cách kết nối các bảng đúng cách; (4) Định nghĩa các chỉ số tính toán một lần và tái sử dụng, đảm bảo tính nhất quán.

Semantic Layer dựa trên Modeling Definition Language (MDL) của WrenAI cho phép định nghĩa Models (các bảng dữ liệu), Views, Relationships (quan hệ giữa các bảng) và Metrics (các chỉ số tính toán). Metadata được serialize thành JSON và lưu trong PostgreSQL, đồng thời được embedding và lưu vào vector database Qdrant để phục vụ semantic search.

2.5. Thiết kế cấp cao (High-level Design)

Thiết kế cấp cao của hệ thống chuyển đổi ngôn ngữ tiếng Việt thành SQL được xây dựng dựa trên kiến trúc Retrieval-Augmented Generation (RAG) chuyên biệt cho Text-to-SQL. Kiến trúc này sử dụng WrenAI Engine làm trung tâm điều phối, tách biệt rõ ràng giữa lớp ngôn ngữ tiếng Việt và lớp dữ liệu vật lý. Thiết kế HLD được tổ chức thành ba lớp logic chính nhằm đảm bảo tính mô-đun hóa, khả năng mở rộng và dễ bảo trì: Lớp Giao diện và Tương tác, Lớp Ứng dụng và Điều phối, và Lớp Trí tuệ và Dữ liệu Cốt lõi.

2.5.1. Lớp Giao diện và Tương tác

Lớp này cung cấp kênh giao tiếp chính giữa người dùng cuối và hệ thống xử lý, tập trung vào trải nghiệm người dùng. Các thành phần bao gồm: Web Portal/Chatbot UI cung cấp giao diện trực quan cho phép người dùng nhập câu hỏi bằng ngôn ngữ

tiếng Việt tự nhiên; Hiển thị Kết quả trình bày kết quả truy vấn dưới dạng dễ hiểu, bao gồm bảng dữ liệu, câu lệnh SQL được sinh ra, và khả năng hiển thị biểu đồ trực quan hóa dữ liệu; Quản lý Tương tác hỗ trợ chức năng lịch sử truy vấn và khả năng điều chỉnh câu hỏi hoặc câu truy vấn SQL khi cần.

2.5.2. Lớp Ứng dụng và Điều phối

Lớp này điều phối toàn bộ luồng xử lý truy vấn, từ tiếp nhận câu hỏi đến xử lý và trả về kết quả, hoạt động như một dịch vụ không lưu trạng thái. Các thành phần bao gồm: API Gateway là điểm tiếp nhận mọi yêu cầu từ Lớp Giao diện; Module Tiền xử lý Ngôn ngữ Việt thực hiện chuẩn hóa câu hỏi và phân tích ý định, trích xuất thực thể (Slot Filling); Dịch vụ Điều phối RAG quản lý toàn bộ quy trình Truy xuất tăng cường sinh, chịu trách nhiệm truy xuất các thành phần cấu trúc liên quan và lắp ráp chúng thành ngữ cảnh cho LLM; Module Xử lý Lỗi và Tự Sửa lỗi áp dụng cơ chế kiểm tra và sửa lỗi tự động cho câu truy vấn SQL được sinh ra, sử dụng vòng lặp phản hồi từ lỗi thực thi.

2.5.3. Lớp Trí tuệ và Dữ liệu Cốt lõi

Lớp này cung cấp tri thức nền tảng và năng lực sinh truy vấn, là thành phần quyết định độ chính xác của hệ thống. Các thành phần bao gồm: Semantic Layer (Tầng Ngữ nghĩa) là lớp trừu tượng hóa, mã hóa logic nghiệp vụ, định nghĩa các chỉ số (Metrics), mối quan hệ (Relationships), và Models, với vai trò quan trọng nhất là ánh xạ thuật ngữ nghiệp vụ tiếng Việt sang cấu trúc kỹ thuật của cơ sở dữ liệu; Vector Store (Qdrant) là cơ sở dữ liệu vector lưu trữ embeddings của schema, ví dụ câu hỏi-SQL, và metadata, được sử dụng cho tìm kiếm ngữ nghĩa (Semantic Search) tìm ngữ cảnh; Mô hình Ngôn ngữ Lớn (LLM) là thành phần trung tâm thực hiện chuyển đổi từ ngôn ngữ tiếng Việt sang câu truy vấn SQL dựa trên ngữ cảnh được cung cấp từ Semantic Layer; Bộ máy Truy vấn Ngữ nghĩa (Query Execution Engine) chịu trách nhiệm kiểm tra tính hợp lệ và an toàn của SQL (ngăn chặn thao tác nguy hiểm như DELETE, DROP), sau đó thực thi câu truy vấn ở chế độ chỉ đọc trên cơ sở dữ liệu nguồn; Cơ sở dữ liệu Nguồn bao gồm PostgreSQL Read-Replica, chứa dữ liệu thực tế của hệ thống đại học số.

2.5.4. Luồng xử lý từ đầu đến cuối

Luồng xử lý một truy vấn ngôn ngữ tiếng Việt (NL) trong kiến trúc RAG diễn ra qua sáu giai đoạn chính: (1) Hiểu Câu hỏi - phân tích câu hỏi NL để xác định ý định, các thực thể và ràng buộc; (2) Truy xuất Cấu trúc - sử dụng tìm kiếm ngữ nghĩa (từ Vector Store) để truy xuất các thành phần cấu trúc (bảng, cột, quan hệ, metrics) liên quan từ Semantic Layer; (3) Lắp ráp Ngữ cảnh - kết hợp cấu trúc đã truy xuất, ví dụ mẫu và quy tắc nghiệp vụ thành câu nhắc toàn diện (Prompt Construction); (4) Sinh SQL - LLM sinh câu truy vấn SQL dựa trên ngữ cảnh đã lắp ráp; (5) Kiểm tra và Sửa lỗi - kiểm tra cú pháp, ngữ nghĩa, và an toàn, áp dụng cơ chế tự sửa lỗi nếu cần; (6) Thực thi và Trả Kết quả - thực thi SQL an toàn trên cơ sở dữ liệu và định dạng kết quả trả về cho người dùng.

2.6. Thiết kế kiến trúc hệ thống

2.6.1. Kiến trúc tổng thể của hệ thống

Hệ thống được thiết kế dựa trên kiến trúc Retrieval-Augmented Generation (RAG) chuyên biệt cho Text-to-SQL. Khác với các mô hình Chatbot thông thường chỉ dựa vào dữ liệu huấn luyện, kiến trúc này sử dụng WrenAI Engine làm trung tâm điều phối, tách biệt hoàn toàn giữa lớp ngôn ngữ tiếng Việt và lớp dữ liệu vật lý. Kiến trúc RAG khắc phục hạn chế của mô hình thuần túy bằng cách bổ sung cơ chế truy xuất động thông tin từ nguồn dữ liệu bên ngoài. Theo nguyên lý hoạt động, thay vì chỉ dựa vào tri thức nội tại của mô hình ngôn ngữ lớn, hệ thống sẽ truy xuất các thông tin liên quan từ kho tri thức để xây dựng ngữ cảnh, sau đó cung cấp ngữ cảnh này cho mô hình nhằm sinh ra câu truy vấn SQL có độ chính xác cao hơn.

2.6.2. Các thành phần cốt lõi của kiến trúc

Dựa trên phân tích từ các công cụ Text-to-SQL hiện đại, một công cụ bao gồm sáu thành phần chính:

Tầng hiểu ngôn ngữ tiếng Việt: Tầng này phân tích làm rõ yêu cầu của người dùng từ câu hỏi. Các chức năng chính bao gồm phân loại ý định truy vấn, nhận diện các thực thể như tên bảng và giá trị cụ thể, phân tích cấu trúc ngữ nghĩa để xác định điều kiện lọc và phép toán. Với mô hình ngôn ngữ lớn, các tác vụ này có thể được thực hiện bằng học vài mẫu hoặc học không mẫu, giảm đáng kể nhu cầu dữ liệu huấn luyện.

Tầng hiểu cấu trúc cơ sở dữ liệu: Đây là thành phần chịu trách nhiệm ánh xạ giữa ngôn ngữ tiếng Việt và cấu trúc cơ sở dữ liệu. Quá trình liên kết cấu trúc ánh xạ các từ trong câu hỏi với các thành phần như bảng, cột, giá trị. Các phương pháp thường được sử dụng bao gồm so khớp trực tiếp, so khớp mờ dựa trên khoảng cách chỉnh sửa, và tìm kiếm tương đồng theo vector nhằm phát hiện quan hệ ngữ nghĩa giữa các thực thể.

Tầng ngữ nghĩa nghiệp vụ: Tầng ngữ nghĩa là một lớp trừu tượng nằm giữa câu hỏi ngôn ngữ tiếng Việt và cấu trúc cơ sở dữ liệu vật lý. Theo nghiên cứu của AtScale (2024) và Denodo (2024), tầng ngữ nghĩa có thể cải thiện độ chính xác từ 20% lên 92.5% khi được triển khai đúng cách. Vai trò bao gồm: mã hóa logic nghiệp vụ; thống nhất thuật ngữ nghiệp vụ, ánh xạ từ thuật ngữ kinh doanh sang tên kỹ thuật; quản lý mối quan hệ giữa các thực thể; định nghĩa các chỉ số tính toán một lần và tái sử dụng.

Tầng truy xuất và lắp ráp ngữ cảnh: Thành phần này thực hiện quy trình truy xuất tăng cường, chịu trách nhiệm truy xuất thông tin liên quan từ kho tri thức và lắp ráp thành ngữ cảnh cho mô hình ngôn ngữ lớn. Quy trình bao gồm: chuyển câu hỏi và các thành phần cấu trúc thành vector; sử dụng cơ sở dữ liệu vector để tìm các thành phần liên quan nhất; xếp hạng lại để cải thiện chất lượng kết quả; ghép nối thông tin đã truy xuất, ví dụ mẫu và hướng dẫn thành một câu nhắc toàn diện.

Tầng sinh SQL: Tầng này sử dụng mô hình ngôn ngữ lớn để sinh câu truy vấn SQL dựa trên ngữ cảnh đã được chuẩn bị. Các phương pháp chính bao gồm: học trong ngữ cảnh bằng cách cung cấp ví dụ mẫu trong câu nhắc; suy luận chuỗi tư duy yêu cầu mô hình giải thích quá trình từng bước trước khi sinh SQL; tinh chỉnh lặp sử dụng vòng lặp tự sửa lỗi, nếu SQL sinh ra có lỗi thì thông báo lỗi được gửi lại cho mô hình để tinh chỉnh.

Tầng kiểm tra và thực thi: Tầng này đảm bảo SQL được sinh ra là an toàn và chính xác trước khi thực thi. Các chức năng bao gồm: kiểm tra cú pháp SQL; xác minh các bảng và cột được tham chiếu tồn tại; đảm bảo SQL không chứa các thao tác nguy hiểm như xóa hoặc sửa dữ liệu; thử thực thi để xác minh tính khả thi; áp dụng các chính sách bảo mật dựa trên quyền người dùng.

2.6.3. Vai trò của cơ sở dữ liệu vector

Cơ sở dữ liệu vector (vector database) là thành phần then chốt trong kiến trúc RAG, đóng vai trò lưu trữ và tìm kiếm thông tin dựa trên độ tương đồng ngữ nghĩa thay vì khớp từ khóa chính xác. Quy trình hoạt động gồm ba giai đoạn: (1) Embedding và indexing - chuyển đổi thông tin (schema, câu hỏi mẫu, documentation) thành vector bằng mô hình embedding và lưu trữ; (2) Semantic search - chuyển câu hỏi người dùng thành vector và tìm các vector gần nhất; (3) Context assembly - tổng hợp kết quả thành context để đưa vào LLM. Các công nghệ phổ biến gồm Pinecone, Weaviate, Qdrant, Chroma, Milvus. Trong WrenAI, hệ thống tích hợp sẵn Qdrant và tự động xử lý việc embedding, indexing và retrieval.

2.7. Khảo sát và đánh giá các công cụ hiện có

Các công cụ Text-to-SQL được phân loại thành hai nhóm: (1) Công cụ chuyển đổi cơ bản - tập trung vào chuyển đổi đơn giản từ câu hỏi thành SQL, người dùng tự thực thi trên hệ thống; (2) Nền tảng trí tuệ kinh doanh tích hợp - cung cấp trải nghiệm chatbot, tự động hiểu cấu trúc dữ liệu, hỗ trợ phân tích và trực quan hóa.

Các công cụ được khảo sát: WrenAI - nền tảng mã nguồn mở với Semantic Layer sử dụng MDL, kiến trúc RAG với Wren Engine xây dựng trên Apache DataFusion, hỗ trợ nhiều nguồn dữ liệu, phù hợp nhất với yêu cầu đề tài. Vanna.AI - framework Python mã nguồn mở, linh hoạt cao nhưng không có Semantic Layer chính thức. Chat2DB - SQL client đa năng tích hợp AI, phù hợp cho kỹ sư dữ liệu. Text2SQL.AI - dịch vụ thương mại cho tác vụ đơn giản.

2.8. Kết luận chương 2

Chương 2 đã hoàn thành mục tiêu nghiên cứu và thiết kế nền tảng cho hệ thống Text-to-SQL. Thông qua nghiên cứu ba khía cạnh quan trọng (chuẩn hóa schema, từ điển dữ liệu, Semantic Layer), thiết kế cấp cao với ba lớp logic, thiết kế kiến trúc tổng thể dựa trên RAG, và đánh giá có hệ thống các công cụ hiện có, chương đã xác định WrenAI là công cụ phù hợp nhất với kiến trúc đã thiết kế.

CHƯƠNG 3: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG

3.1. Môi trường và công nghệ triển khai

Hệ thống được xây dựng trên nền tảng WrenAI với các công nghệ và công cụ chính:

Môi trường phát triển: Docker + Docker Compose để container hóa ứng dụng, đảm bảo tính nhất quán môi trường; Ubuntu Linux 24.04 LTS làm hệ điều hành triển khai; Visual Studio Code làm IDE phát triển.

Backend và xử lý dữ liệu: Python 3.11+ là ngôn ngữ lập trình chính xử lý AI Service; FastAPI làm web framework xây dựng RESTful API; PostgreSQL lưu trữ metadata, schema và từ điển; Qdrant làm vector database cho semantic search; Wren Engine xử lý semantic mapping.

3.2. Xây dựng các tài nguyên nền tảng

3.2.1. Xây dựng và chuẩn hóa schema cơ sở dữ liệu

Hệ thống sử dụng cơ sở dữ liệu PostgreSQL 15 để quản lý dữ liệu đào tạo của Học viện Công nghệ Bưu chính Viễn thông. Cấu trúc bao gồm hơn 200 bảng chính với hơn 3000 thuộc tính, được tổ chức thành các nhóm chức năng: quản lý tổ chức (khóa sinh viên, ngành, chương trình đào tạo), quản lý sinh viên, quản lý học vụ (học phần, lớp học phần, đăng ký, điểm).

3.2.2. Xây dựng bộ từ điển dữ liệu

Bộ từ điển được lưu trữ trong bảng riêng trong PostgreSQL với cấu trúc gồm: tên bảng, tên thuộc tính, kiểu dữ liệu, mô tả tiếng Việt, ghi chú tham chiếu về vùng giá trị, và metadata bổ sung như độ dài dữ liệu, cho phép null, khóa chính, khóa ngoại, enum.

Quy trình xây dựng qua 3 bước: (1) Thu thập thuật ngữ từ schema bằng script Python tự động trích xuất từ information_schema; (2) Bổ sung từ tài liệu nghiệp vụ như quy chế đào tạo, văn bản hành chính, biểu mẫu; (3) Validate và kiểm tra xung đột các thuật ngữ trùng lặp hoặc đa nghĩa, xử lý bằng cách thêm context vào description.

3.2.3. Xây dựng bộ câu hỏi mẫu

Bộ câu hỏi mẫu là tập hợp các cặp (câu hỏi tiếng Việt, SQL query tương ứng) được đưa vào context của LLM khi xử lý. Kỹ thuật few-shot learning cho phép mô hình học cách chuyển đổi thông qua ví dụ cụ thể, cải thiện độ chính xác với các trường hợp phức tạp hoặc đặc thù.

3.3. Triển khai các module xử lý chính

3.3.1. Module tiền xử lý câu hỏi

Module tiền xử lý thực hiện chuẩn hóa văn bản, phân tích ý định người dùng và trích xuất thực thể. Quá trình bao gồm: chuẩn hóa Unicode, xử lý viết tắt và từ đồng nghĩa, tách từ tiếng Việt, nhận diện các thực thể như tên, mã số, khoảng thời gian.

3.3.2. Module ánh xạ schema

Module ánh xạ schema (Schema Linking) nhận diện và liên kết các thành phần ngôn ngữ trong câu hỏi với các bảng và cột trong cơ sở dữ liệu. Quá trình sử dụng kết hợp: so khớp trực tiếp dựa trên tương đồng từ khóa, so khớp mờ dựa trên khoảng cách chỉnh sửa, và tìm kiếm tương đồng theo vector qua semantic search.

3.3.3. Module tương tác với mô hình ngôn ngữ lớn

Module này xây dựng prompt tối ưu kết hợp thông tin schema và từ điển, gửi tới GPT-4 API để sinh SQL. Quy trình bao gồm: thiết lập kết nối API với authentication và các tham số điều khiển; xây dựng cơ chế xử lý lỗi với retry logic, timeout handling và fallback mechanism; parsing và validation output để extract SQL, validate cú pháp trước khi thực thi.

3.3.4. Module kiểm tra và thực thi SQL

Module thực hiện kiểm tra an toàn trước khi thực thi: syntax validation phát hiện lỗi cú pháp sử dụng SQL parser; security validation kiểm tra các lệnh nguy hiểm như DROP, DELETE, TRUNCATE; schema validation xác minh tên bảng và cột tồn tại. Hệ thống áp dụng cơ chế tự sửa lỗi với vòng lặp phản hồi.

3.4. Xây dựng giao diện người dùng

Giao diện nhập câu hỏi: Thiết kế đơn giản với ô nhập văn bản hỗ trợ tiếng Việt có dấu và không dấu, nút gửi câu hỏi, hiển thị lịch sử truy vấn ở thanh bên. Giao diện được thiết kế theo nguyên tắc lấy người dùng làm trung tâm, hướng tới người dùng không biết SQL.

Giao diện hiển thị kết quả: Tổng hợp toàn bộ quy trình Text-to-SQL với 3 tab chuyển đổi: (1) Câu trả lời - hiển thị mặc định kết quả truy vấn dạng bảng; (2) Xem câu truy vấn - hiển thị SQL cuối cùng để người dùng kiểm tra và có thể lưu lại làm câu hỏi mẫu; (3) Biểu đồ - trực quan hóa dữ liệu theo yêu cầu người dùng.

3.5. Thử nghiệm và đánh giá hệ thống

3.5.1. Kế hoạch và phương pháp thử nghiệm

Hệ thống áp dụng kiểm thử theo kim tự tháp với ba tầng: kiểm thử từng thành phần, kiểm thử kết hợp giữa các thành phần, và kiểm thử toàn hệ thống. Hệ thống được kiểm tra trên bộ gồm 1139 câu hỏi với 3 mức độ: Mức độ dễ - truy vấn một bảng; Mức độ trung bình - truy vấn 2 bảng; Mức độ khó - truy vấn 3 bảng.

3.5.2. Đánh giá độ chính xác của hệ thống

Với bộ câu hỏi mẫu 1139 câu, hệ thống đạt Execution Accuracy là 78.2%, nghĩa là khoảng 890 câu hỏi được chuyển thành câu truy vấn thực hiện đúng. Kết quả cho thấy hệ thống đáp ứng được yêu cầu về độ chính xác đặt ra (75-80%), chứng minh tính khả thi của mô hình Text-to-SQL với tiếng Việt.

3.5.3. Đánh giá hiệu suất của hệ thống

Kiểm thử tải được thực hiện bằng công cụ artillery với các kịch bản 20, 50, 100, 200 người dùng đồng thời. Kết quả cho thấy: với 20 người dùng, thời gian phản hồi trung bình 7.6ms, tỷ lệ thành công 100%; với 50 người dùng, thời gian phản hồi 1616.7ms, tỷ lệ thành công 100%; với 100 người dùng, thời gian phản hồi 12179.1ms, tỷ lệ thành công 100%; với 200 người dùng, thời gian phản hồi 25384ms, tỷ lệ thành công giảm còn 70.85% với tỷ lệ lỗi 29.15%.

Kết quả cho thấy hệ thống hoạt động ổn định với 20-100 người dùng đồng thời. Khi tải tăng lên 200 người dùng, tỷ lệ lỗi tăng đáng kể, cho thấy cần các giải pháp tối ưu để mở rộng quy mô.

3.6. Cải tiến và tối ưu hóa

3.6.1. Phân tích điểm yếu của hệ thống

Các hạn chế được nhận diện: Độ chính xác còn hạn chế với câu hỏi phức tạp - hệ thống gặp khó khăn khi câu hỏi có cấu trúc nhiều mệnh đề, điều kiện lồng nhau, nhiều ràng buộc. Một số SQL sinh ra đúng cú pháp nhưng trả về dữ liệu sai so với ý định. Tốc độ phản hồi không ổn định với bảng dữ liệu lớn.

3.6.2. Các giải pháp cải tiến

Cải tiến độ chính xác: Bổ sung thêm mô tả chi tiết cho schema, làm giàu từ điển dữ liệu với các biến thể cách diễn đạt, tăng số lượng câu hỏi mẫu cho few-shot learning, tinh chỉnh prompt để hướng dẫn LLM xử lý các trường hợp phức tạp.

Cải tiến hiệu năng: Áp dụng cơ chế cache cho các truy vấn phổ biến, tối ưu index trên cơ sở dữ liệu, giới hạn số lượng kết quả trả về, sử dụng connection pooling để quản lý kết nối hiệu quả.

3.7. Triển khai và hướng dẫn sử dụng

3.7.1. Hướng dẫn cài đặt và cấu hình

Quy trình triển khai: Bước 1 - Cài đặt Docker phiên bản mới nhất trên máy chủ, tải mã nguồn từ GitHub và thực hiện cài đặt bằng docker compose. Bước 2 - Sau khi server khởi tạo thành công, mở giao diện Wren UI và cấu hình kết nối đến cơ sở dữ liệu nguồn, chỉ cung cấp username với quyền đọc dữ liệu để đảm bảo an toàn.

3.7.2. Hướng dẫn sử dụng cho người dùng cuối

Giao diện người dùng được thiết kế đơn giản hướng tới người dùng không biết SQL: (1) Truy cập giao diện Wren AI UI trên trình duyệt; (2) Nhập câu hỏi bằng tiếng Việt vào khung chat; (3) Nhấn gửi và chờ hệ thống sinh SQL và trả về kết quả; (4)

Kiểm tra kết quả, xem báo cáo hoặc biểu đồ; (5) Nếu kết quả lỗi, cải thiện câu hỏi hoặc chỉnh sửa câu truy vấn.

3.7.3. Hướng dẫn vận hành và bảo trì

Để hệ thống hoạt động ổn định cần: (1) Kiểm tra và bảo trì định kỳ - theo dõi các truy vấn sinh ra, lỗi cú pháp, hiệu năng, cập nhật schema khi cơ sở dữ liệu thay đổi, sao lưu dữ liệu định kỳ; (2) Giám sát hiệu suất - theo dõi thời gian thực thi trung bình, số lượng kết nối, điều chỉnh cấu hình; (3) Cập nhật và tinh chỉnh model, prompt khi dữ liệu thay đổi.

3.8. Kết luận chương 3

Chương 3 đã trình bày một cách hệ thống quy trình triển khai hệ thống chuyển đổi ngôn ngữ tiếng Việt sang câu truy vấn SQL dựa trên nền tảng WrenAI tích hợp với PostgreSQL 15. Thông qua việc mô tả kiến trúc tổng thể, cơ chế tích hợp giữa các thành phần, và phương pháp phát triển giao diện người dùng theo nguyên tắc lấy người dùng làm trung tâm, nghiên cứu đã làm rõ cách thức hoạt động của từng module.

Kết quả thử nghiệm với 1139 câu hỏi đạt Execution Accuracy 78.2%, chứng minh tính khả thi của mô hình Text-to-SQL với tiếng Việt. Hệ thống hoạt động ổn định với 20-100 người dùng đồng thời. Các hoạt động triển khai và tài liệu hướng dẫn kỹ thuật đã được hệ thống hóa, tạo điều kiện cho việc nhân rộng giải pháp trong thực tế.

KẾT LUẬN VÀ KIẾN NGHỊ

1. Kết luận

Đề tài đã đạt được mục tiêu ban đầu trong việc thiết kế và triển khai một công cụ vận dụng trí tuệ nhân tạo nhằm thực hiện chuyển đổi tự động từ câu hỏi bằng tiếng Việt sang câu truy vấn SQL, từ đó mang lại một giải pháp truy xuất thông tin có tính ứng dụng cao cho bối cảnh Đại học số ở Việt Nam.

Song song với những thành tựu đạt được, công trình nghiên cứu vẫn còn tồn tại một số hạn chế. Về độ chính xác, hệ thống gặp khó khăn khi xử lý các câu truy vấn phức tạp cao, đặc biệt là những truy vấn đòi hỏi liên kết nhiều nguồn dữ liệu, sử dụng truy vấn lồng nhau và thực hiện các phép tính thống kê nâng cao. Nguyên nhân xuất phát từ việc mô hình ngôn ngữ chưa trải qua quá trình huấn luyện chuyên sâu dành riêng cho lĩnh vực giáo dục với ngữ liệu tiếng Việt. Về triển khai thực tế, việc phụ thuộc vào API của các mô hình ngôn ngữ thương mại đặt ra thách thức về chi phí vận hành và vấn đề bảo mật khi cần truyền tải thông tin đến các bên cung cấp dịch vụ bên ngoài.

Dựa trên những hạn chế đã nhận diện, nghiên cứu đề xuất các định hướng phát triển: triển khai huấn luyện bổ sung cho mô hình với tập ngữ liệu tiếng Việt chuyên ngành giáo dục, xây dựng hệ thống từ điển thuật ngữ phong phú hơn, thiết kế cơ chế xử lý ngữ cảnh tiên tiến. Việc tích hợp phương pháp học tăng cường dựa trên phản hồi người sử dụng, kết hợp với module tự động phát hiện và điều chỉnh lỗi trong truy vấn, sẽ giúp hệ thống tự hoàn thiện qua thời gian. Về mở rộng chức năng, hệ thống có tiềm năng được bổ sung năng lực phân tích dự báo, hệ thống khuyến nghị thông minh, cùng khả năng tạo lập biểu đồ và bảng điều khiển tương tác để hỗ trợ đưa ra quyết định dựa trên dữ liệu được trực quan hóa.