

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**



VÕ LÊ THÀNH PHÁT

**KHOÁ LUẬN TỐT NGHIỆP
EDUTWIN - BẢN SAO HỌC TẬP KỸ THUẬT SỐ**

EDUTWIN - EDUCATION DIGITAL TWIN

KỸ SỰ NGÀNH MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU

TP. HỒ CHÍ MINH, 2025

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**



VÕ LÊ THÀNH PHÁT – 21522452

**KHOÁ LUẬN TỐT NGHIỆP
EDUTWIN - BẢN SAO HỌC TẬP KỸ THUẬT SỐ
EDUTWIN - EDUCATION DIGITAL TWIN**

KỸ SỰ NGÀNH MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU

**GIẢNG VIÊN HƯỚNG DẪN
TS. Nguyễn Tân Hoàng Phước**

TP. HỒ CHÍ MINH, 2025

THÔNG TIN HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số ngày ...
..... của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. Chủ tịch:
2. Thư ký:
3. Ủy viên:

LỜI CẢM ƠN

Từ thời điểm tìm kiếm đến khi thực hiện xong bài luận văn, tôi rất biết ơn tất cả sự hướng dẫn, động viên và sự hỗ trợ từ phía Nhà trường, Thầy Cô, bạn bè, đồng nghiệp và Gia đình.

Tôi xin gửi lời chân thành cảm ơn đến Thầy hướng dẫn là **TS. Nguyễn Tân Hoàng Phước** đã nhiệt tình chỉ dẫn trong suốt thời gian tìm hiểu, nghiên cứu thực nghiệm cho đến thời điểm hoàn thành bài luận văn tại trường.

Tôi chân thành gửi lời tri ân đến các **Quý Thầy Cô trong Phòng Đào tạo của trường Đại học Công nghệ Thông tin** đã hỗ trợ cũng như cung cấp cho tôi những tri thức, kiến thức, hướng dẫn các thủ tục và các kinh nghiệm quý giá cho tôi trong suốt quá trình học tập nghiên cứu ở trường.

Đồng thời, tôi cũng muốn gửi những lời cảm ơn chân thành tới Gia đình, bạn bè và các đồng nghiệp đã hỗ trợ tôi trong suốt toàn bộ quá trình học và thực nghiệm nhằm hoàn thành nội dung bài luận văn này.

Với khoảng thời gian ngắn đồng thời kiến thức bản thân tôi còn có nhiều hạn chế, do đó bài luận văn chắc chắn vẫn còn các thiếu sót. Tôi rất mong sẽ nhận được những lời khuyên, góp ý của quý Thầy Cô.

Nhóm tác giả

TÓM TẮT

Trong bối cảnh chuyển đổi số giáo dục, việc cá nhân hóa trải nghiệm học tập đang trở thành yêu cầu cấp thiết. Tuy nhiên, các hệ thống quản lý học tập hiện tại thường chỉ dừng lại ở việc lưu trữ kết quả mà thiếu đi khả năng phân tích và dự báo. Khóa luận này trình bày việc nghiên cứu và xây dựng "**EduTwin - Bản sao Học tập Kỹ thuật số tích hợp AI**", một giải pháp nhằm khắc phục các hạn chế trên.

EduTwin ứng dụng ý tưởng "**Digital Twin**" để tạo ra một bản sao kỹ thuật số của người học, có khả năng "tiến hóa" thông qua dữ liệu tương tác liên tục. Hệ thống kết hợp thuật toán **Lazy Learning** (K-NN, KR, LWLR) để dự đoán kết quả học tập tức thời mà không cần huấn luyện lại, cùng với các **Large Language Model** (LLM) để đóng vai trò như một trợ lý ảo thấu hiểu ngữ cảnh và tâm lý người học. Kết quả thực nghiệm cho thấy hệ thống không chỉ cung cấp các dự báo có độ tin cậy cao mà còn tạo ra trải nghiệm tương tác tự nhiên, giúp học sinh chủ động hơn trong việc định hướng lộ trình học tập của mình.

MỤC LỤC

Thông tin hội đồng bảo vệ khóa luận	i
Lời cảm ơn	ii
Tóm tắt	iii
Mục lục	iv
Danh sách bảng	viii
Danh sách hình vẽ	ix
Danh mục từ viết tắt	xii
Chương 1. TỔNG QUAN VỀ ĐỀ TÀI	1
1.1 Lý do chọn đề tài	1
1.2 Mục tiêu, đối tượng và phạm vi nghiên cứu	2
1.2.1 Mục tiêu nghiên cứu	2
1.2.2 Đối tượng nghiên cứu	3
1.2.3 Phạm vi nghiên cứu	4
1.2.3.1 Phạm vi về Dữ liệu và Ngữ cảnh áp dụng	4
1.2.3.2 Phạm vi về Chức năng hệ thống	5
1.2.3.3 Phạm vi về Công nghệ	5
1.3 Phương pháp nghiên cứu	6
1.4 Tổng quan đề tài	7
1.4.1 Các nghiên cứu và giải pháp liên quan	7
1.4.1.1 Digital Twin trong Giáo dục: Từ mô phỏng môi trường đến bản sao người học	8
1.4.1.2 Học tập thích ứng và Bài toán dự báo hiệu suất: Hướng tiếp cận Lazy Learning	8
1.4.1.3 Generative Artificial Intelligence (AI) và Large Language Model (LLM): Từ công cụ tạo sinh đến trợ lý ngữ cảnh	9
1.4.1.4 Xử lý dữ liệu khuyết thiêу trong hồ sơ học tập	10
1.4.2 Đóng góp mới của đề tài	10

MỤC LỤC

1.5 Câu trúc Khoa luận tốt nghiệp	12
Chương 2. Cơ sở lý thuyết	13
2.1 Digital Twin và Học tập Cá nhân hóa	13
2.2 Lazy Learning	14
2.3 Phân cụm K-Means và Tối ưu hóa Mẫu hình	15
2.3.1 Thuật toán K-Means Clustering	16
2.3.2 Xác định số lượng cụm tối ưu	17
2.3.3 Kiến trúc hai tầng (2-Tier Architecture)	17
2.3.4 Quy trình suy diễn (Inference Pipeline)	17
2.3.5 Kết quả thực nghiệm	18
2.4 Kiến trúc Transformer và Cơ chế Attention.	18
2.5 Xử lý Dữ liệu và Kỹ thuật Imputation	19
2.6 Kiến trúc AI Agent và Mẫu thiết kế ReAct	19
2.7 Các chỉ số Đánh giá Hiệu năng Mô hình	20
2.7.1 Sai số tuyệt đối trung bình (MAE - Mean Absolute Error)	20
2.7.2 Căn bậc hai sai số bình phương trung bình (RMSE - Root Mean Squared Error)	20
2.7.3 Hệ số xác định (R^2 - R-squared)	21
2.8 Trực quan hóa Dữ liệu Giáo dục.	21
2.9 Xác thực API với Session-based Authentication	22
Chương 3. Phân tích và Thiết kế hệ thống	23
3.1 Phát biểu bài toán	23
3.1.1 Bối cảnh	23
3.1.2 Bài toán cần giải quyết	23
3.1.3 Các thách thức kỹ thuật và Giải pháp	23
3.2 Quy trình thực hiện	24
3.3 Phân tích yêu cầu hệ thống	25
3.3.1 Yêu cầu chức năng	25
3.3.2 Yêu cầu phi chức năng	29
3.4 Mô hình hóa quy trình nghiệp vụ	30
3.4.1 Biểu đồ Use Case	30
3.4.2 Quy trình Đăng nhập và Xác thực	31

MỤC LỤC

3.4.3	Quy trình Dự đoán Điểm số	31
3.4.4	Quy trình Chat với AI (Chatbot Mode)	32
3.4.5	Quy trình Learning Agent (Learning Mode)	32
3.4.6	Quy trình Thiết lập Cấu trúc Học tập	33
3.4.7	Quy trình Upload Dataset và Đánh giá Model	33
3.4.8	Quy trình Upload Tài liệu Học tập (User)	34
3.4.9	Kiến trúc tổng thể	34
3.4.10	Kiến trúc Module AI	34
3.5	Thiết kế Cơ sở dữ liệu	35
3.5.1	Thiết kế mức khái niệm	36
3.5.2	Thiết kế chi tiết lược đồ dữ liệu	36
3.5.3	Phân tích các quyết định thiết kế quan trọng	40
3.6	Thiết kế Thuật toán và Luồng xử lý dữ liệu	43
3.6.1	Luồng xử lý dữ liệu khuyết	44
3.6.2	Chiến lược Tối ưu hóa Dự đoán với Phân cụm K-Means và Tuyển chọn Mẫu hình	44
3.6.3	Luồng dự đoán Lazy Learning	45
3.6.4	Luồng xử lý AI Chatbot (Chat Mode)	47
3.6.5	Luồng xử lý Learning Agent (Learning Mode)	48
3.7	Thiết kế Giao diện người dùng	49
3.7.1	Sơ đồ tổ chức thông tin	49
3.7.2	Thiết kế chi tiết các màn hình chính	49
3.8	Tổng hợp Công nghệ và Môi trường Triển khai	51
3.8.1	Danh sách Công nghệ	52
3.8.2	Môi trường Triển khai	53
Chương 4.	Thực nghiệm và đánh giá	54
4.1	Môi trường và Dữ liệu thực nghiệm	54
4.1.1	Môi trường thực nghiệm	54
4.1.1.1	Phân tích Khám phá Dữ liệu	54
4.2	Kịch bản 1: Đánh giá hiệu quả xử lý dữ liệu khuyết	56
4.2.1	Thiết lập thực nghiệm	56
4.2.2	Phân tích kết quả	56

MỤC LỤC

4.3 Kịch bản 2: Đánh giá hiệu năng đa mô hình — So sánh Lazy Learning và Eager Learning	58
4.3.1 Kết quả thực nghiệm định lượng	59
4.3.2 Phân tích kết quả	59
4.4 Kịch bản 3: Đánh giá hiệu năng và Tốc độ phản hồi	60
4.4.1 Thiết lập thực nghiệm và Dữ liệu	60
4.4.2 Phân tích kết quả thực nghiệm.	60
4.5 Kịch bản 4: Đánh giá Chatbot và Bảo mật Personally Identifiable Information (PII)	61
4.5.1 Kiểm thử Personally Identifiable Information (PII) Redaction (Bảo mật)	61
4.5.2 Kiểm thử Cá nhân hóa với Context Injection	61
4.6 Kịch bản 5: Đánh giá Learning Agent	63
4.6.1 Kiểm thử Tool Usage	63
4.6.2 Kiểm thử Multi-step Reasoning	63
4.7 Bàn luận chung	64
4.8 Kết quả xây dựng ứng dụng	65
4.8.1 Giao diện đăng nhập/đăng ký	65
4.8.2 Phân hệ Quản trị viên	66
4.8.3 Phân hệ Người dùng	66
Chương 5. Kết luận và hướng phát triển	74
5.1 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	74
5.1.1 Kết luận chung	74
5.1.2 Các hạn chế của đề tài	75
5.1.3 Hướng phát triển	75
Phụ lục A. Các thông tin liên quan	77
A.1 Tài nguyên thực nghiệm	77
Tài liệu tham khảo	80

DANH SÁCH BẢNG

3.1	Các thách thức kỹ thuật và Giải pháp đề xuất	24
3.2	Danh sách chức năng cho Học sinh	26
3.3	Danh sách chức năng cho Admin	28
3.4	Danh sách chức năng của AI Core	29
3.5	Bảng custom_teaching_structures - Định nghĩa cấu trúc học tập .	37
3.6	Bảng custom_user_scores - Lưu trữ điểm số người dùng	38
3.7	Bảng custom_reference_dataset - Dữ liệu mẫu (Training Data) .	38
3.8	Bảng users - Thông tin người dùng	39
3.9	Bảng ai_insights - Lưu trữ nhận định AI về học sinh	40
3.10	Bảng custom_structure_documents - Lưu tài liệu tham khảo cho AI	41
3.11	Bảng ml_model_parameters - Tham số các mô hình ML	41
3.12	Bảng ml_model_config - Cấu hình mô hình ML đang hoạt động . . .	42
3.13	Bảng chat_sessions - Quản lý phiên hội thoại	42
3.14	Bảng chat_messages - Lưu trữ tin nhắn	42
3.15	Bảng user_structure_preferences - Lưu trữ preferences của user theo structure	43
4.1	So sánh hiệu năng điền khuyết (KNN so với các phương pháp thống kê) .	58
4.2	So sánh hiệu năng giữa các mô hình Lazy Learning và Eager Learning .	59
4.3	Kết quả Stress Test so sánh hiệu năng (dữ liệu thực tế, 54 features)	61
4.4	Hiệu quả của module Personally Identifiable Information (PII) Redaction .	62
4.5	So sánh Prompt và Phản hồi giữa các mức độ Context	62
4.6	Kết quả kiểm thử Tool Usage	63

DANH SÁCH HÌNH VẼ

DANH SÁCH HÌNH VẼ

4.13 Giao diện Learning Agent với hiển thị tiền trình suy luận ReAct	72
4.14 Giao diện Cài đặt với danh sách sở thích đã học (Learned Preferences) . . .	73

DANH MỤC TỪ VIẾT TẮT

AI	Artificial Intelligence
API	Application Programming Interface
EAV	Entity–Attribute–Value
ERD	Entity Relationship Diagram
GIN	Generalized Inverted Index
GPA	Grade Point Average
GPU	Graphics Processing Unit
IBL	Instance-based Learning
IoT	Internet of Things
ITS	Intelligent Tutoring Systems
KNN	k-Nearest Neighbors
KNNI	k-Nearest Neighbors Imputation
KR	Kernel Regression
LLM	Large Language Model
LMS	Learning Management System
LSTM	Long Short-Term Memory
LWLR	Locally Weighted Linear Regression
MAE	Mean Absolute Error
MCAR	Missing Completely At Random
ML	Machine Learning
MVP	Minimum Viable Product
PII	Personally Identifiable Information
RAG	Retrieval-Augmented Generation
RMSE	Root Mean Squared Error
SOA	Service-Oriented Architecture
THPT	Trung Học Phổ Thông
TTL	Time To Live
UML	Unified Modeling Language
ZPD	Zone of Proximal Development

Chương 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1 Lý do chọn đề tài

Trong bối cảnh Cách mạng công nghiệp 4.0, chuyển đổi số trong giáo dục đã trở thành một xu thế tất yếu trên toàn cầu. Tại Việt Nam, Quyết định số 131/QĐ-TTg [1] của Thủ tướng Chính phủ đã phê duyệt Đề án "Tăng cường ứng dụng công nghệ thông tin và chuyển đổi số trong giáo dục và đào tạo giai đoạn 2022-2025", đặt nền móng cho việc ứng dụng các công nghệ tiên tiến như Trí tuệ nhân tạo (Artificial Intelligence (AI)) và Dữ liệu lớn (Big Data). Tuy nhiên, thực trạng các hệ thống giáo dục hiện nay vẫn còn tồn tại nhiều bất cập đáng kể.

Thứ nhất, các hệ thống quản lý học tập (Learning Management System (LMS)) truyền thống như VnEdu (VNPT), SMAS (Viettel), Google Classroom (Google) chủ yếu đóng vai trò là "kho lưu trữ thụ động". Chúng làm tốt nhiệm vụ số hóa điểm số và hồ sơ, nhưng hoàn toàn thiếu vắng khả năng phân tích sâu (deep analytics) để đưa ra các dự báo mang tính chiến lược. Học sinh và phụ huynh thường chỉ biết kết quả khi kỳ thi đã kết thúc, dẫn đến việc mọi biện pháp can thiệp đều trở nên muộn màng. Sự thiếu hụt thông tin định hướng khiến người học rơi vào trạng thái "mù mờ" về năng lực thực sự và lộ trình phát triển của bản thân.

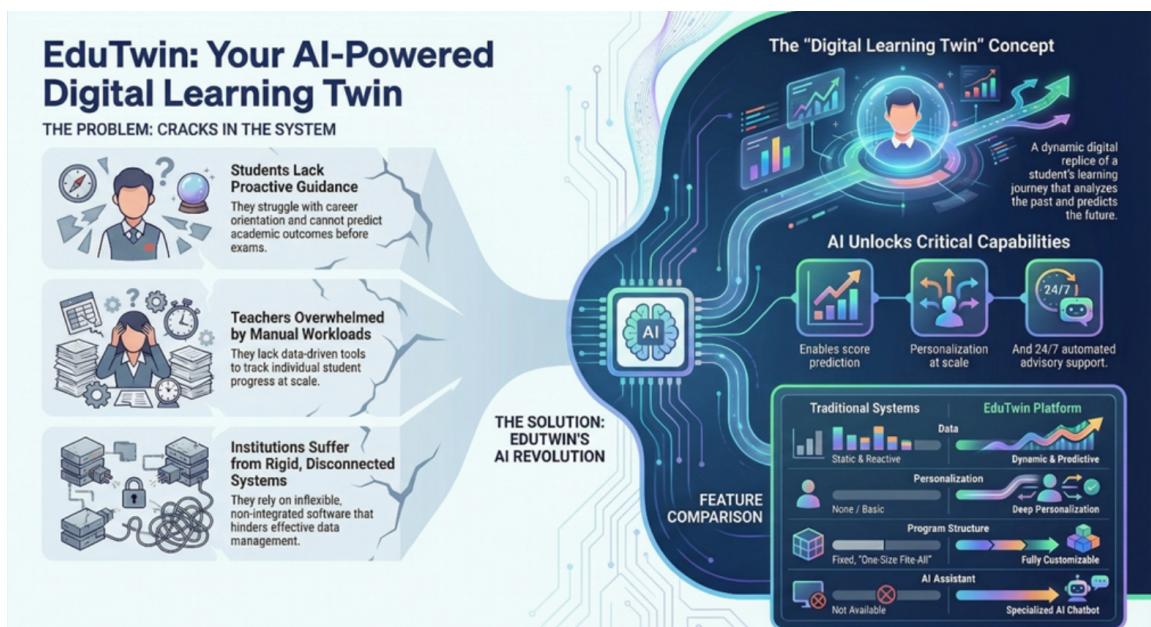
Thứ hai, khái niệm "Cá nhân hóa học tập" (Personalized Learning) thường chỉ dừng lại ở khẩu hiệu hoặc các tính năng cơ bản như gợi ý bài tập dựa trên quy tắc cứng. Chưa có một hệ thống nào thực sự thấu hiểu hành vi, thói quen và phong cách học tập của từng cá nhân để đóng vai trò như một người bạn đồng hành thực thụ.

Xuất phát từ nhu cầu cấp thiết đó, đề tài "EduTwin - Bản sao Học tập Kỹ thuật số tích hợp AI" được đề xuất nghiên cứu và phát triển. Ý tưởng cốt lõi của EduTwin là khởi tạo một "Bản sao số" (Digital Twin) cho mỗi học sinh. Khác với các hồ sơ tĩnh truyền thống, bản sao số này là một thực thể sống động, có khả năng tiến hóa cùng với sự phát triển của người học.

EduTwin vận hành dựa trên một vòng lặp tương tác thông minh và liên tục:

- **Thu thập dữ liệu đa chiều:** Hệ thống không chỉ ghi nhận điểm số mà còn thẩm thấu các dữ liệu hành vi, thói quen và sở thích thông qua quá trình tương tác tự nhiên với người dùng.
- **Phân tích & Dự báo thời gian thực:** Ứng dụng các thuật toán Lazy Learning, hệ

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI



Hình 1.1. Mô hình tổng quan EduTwin: Giải pháp khắc phục các hạn chế về dự báo và cá nhân hóa của hệ thống giáo dục truyền thống thông qua công nghệ Digital Learning Twin.

thống có khả năng dự báo kết quả học tập ngay lập tức (real-time prediction) mà không cần quy trình huấn luyện lại tốn kém, giúp người học nhận diện sớm các nguy cơ hoặc cơ hội.

- **Thích nghi & Đồng hành:** Thông qua Chatbot tích hợp Large Language Model (LLM), EduTwin tự động điều chỉnh phong cách giao tiếp, từ ngôn ngữ trang trọng đến thân mật, từ nghiêm khắc đến khích lệ, để phù hợp nhất với tâm lý của từng học sinh.

Đây chính là sự chuyển dịch mô hình từ "công cụ hỗ trợ thụ động" sang "trợ lý thông minh chủ động", đánh dấu một bước tiến mới trong việc ứng dụng công nghệ để khai phóng tiềm năng con người.

1.2 Mục tiêu, đối tượng và phạm vi nghiên cứu

1.2.1 Mục tiêu nghiên cứu

Mục tiêu tổng quát của đề tài là thiết kế và hiện thực hóa nền tảng EduTwin - một hệ sinh thái giáo dục thông minh, linh hoạt và lấy người học làm trung tâm. Hệ thống được xây dựng để giải quyết bài toán cá nhân hóa ở mức độ sâu (deep personalization) và cung cấp khả năng tùy biến cao cho các cơ sở giáo dục.

Các mục tiêu cụ thể bao gồm:

- **Xây dựng Engine dự đoán thích ứng nhanh và hiệu năng cao:** Nghiên cứu và triển khai nhóm thuật toán Lazy Learning (k-Nearest Neighbors (KNN), Kernel Regression (KR), Locally Weighted Linear Regression (LWLR)). Đặc biệt, giải quyết bài toán độ trễ tính toán trên dữ liệu lớn bằng chiến lược truy vấn hai tầng: kết hợp **Phân cụm đánh chỉ mục (Cluster Indexing)** và **Tuyển chọn mẫu hình thích ứng (Adaptive Prototype Selection)**. Điều này đảm bảo hệ thống duy trì độ phức tạp tính toán thấp, đạt tốc độ phản hồi thời gian thực (Real-time) ngay cả khi quy mô dữ liệu mở rộng.
- **Phát triển Trợ lý ảo AI thấu hiểu ngữ cảnh (Context-Aware AI):** Tích hợp các mô hình ngôn ngữ lớn (LLM) tiên tiến để xây dựng một Chatbot không chỉ trả lời câu hỏi mà còn có khả năng ghi nhớ lịch sử hội thoại, thấu hiểu ngữ cảnh giáo dục và tự động điều chỉnh phản hồi dựa trên hồ sơ tâm lý của học sinh.
- **Phát triển Agent học tập tự động (Learning Agent):** Xây dựng AI Agent sử dụng kiến trúc ReAct (Reasoning + Acting) kết hợp Self-Reflection, cho phép Agent: Tự động suy luận và chọn công cụ phù hợp, Tự đánh giá chất lượng câu trả lời trước khi phản hồi.
- **Kiến trúc Cấu trúc tùy biến (Dynamic Custom Structure):** Giải quyết bài toán "cứng nhắc" của các phần mềm hiện có bằng cách cho phép Quản trị viên (Admin) tự định nghĩa hoàn toàn cấu trúc dữ liệu (môn học, hệ số, quy tắc đánh giá) thông qua giao diện. Điều này giúp EduTwin có thể áp dụng cho mọi mô hình giáo dục, từ Trung Học Phổ Thông (THPT) công lập, trường tư thục quốc tế đến các trung tâm luyện thi chứng chỉ.

1.2.2 Đối tượng nghiên cứu

Để đạt được các mục tiêu trên, đề tài tập trung nghiên cứu vào các đối tượng sau:

- **Dữ liệu học tập số hóa:** Bao gồm dữ liệu điểm số có cấu trúc (structured grades), dữ liệu phi cấu trúc từ lịch sử hội thoại (chat logs), các metadata về cấu trúc chương trình học, và tài liệu học tập do người dùng tải lên (PDF, DOCX, TXT).
- **Thuật toán Học lười và Tối ưu hóa:** Tập trung sâu vào các kỹ thuật KNN, KR và LWLR. Đồng thời nghiên cứu kỹ thuật Phân cụm KMeans (K-Means Clustering) kết

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

hợp Tuyển chọn mẫu hình thích ứng (Adaptive Prototype Selection) để tối ưu không gian tìm kiếm và giảm độ phức tạp tính toán từ $O(N)$ xuống $O(N/K)$.

- **Kiến trúc Agent thông minh:** Nghiên cứu mẫu thiết kế ReAct (Reasoning + Acting) kết hợp Self-Reflection để xây dựng AI Agent có khả năng suy luận nhiều bước, sử dụng công cụ (Tool Use), và tự đánh giá chất lượng phản hồi trước khi trả lời.
- **Kỹ thuật Prompt Engineering & Context Management:** Nghiên cứu các phương pháp tối ưu hóa ngữ cảnh (context window) và che giấu thông tin định danh (PII Redaction) để tích hợp an toàn các mô hình LLM vào môi trường giáo dục.
- **Phương pháp cá nhân hóa Hybrid:** Kết hợp kỹ thuật nhận diện từ khóa (Keyword Detection) với phân tích LLM để học sở thích người dùng một cách hiệu quả về chi phí API.

Khách hàng mục tiêu và người thụ hưởng của hệ thống:

- **Học sinh/Sinh viên:** Những người cần một lộ trình định hướng rõ ràng và một người bạn đồng hành 24/7.
- **Nhà quản lý giáo dục (Admin/Manager):** Những người cần một công cụ linh hoạt để quản lý chất lượng đào tạo theo các tiêu chuẩn riêng biệt của cơ sở mình.

1.2.3 Phạm vi nghiên cứu

Để đảm bảo tính khả thi trong khuôn khổ thời gian của khóa luận, phạm vi nghiên cứu được xác định cụ thể trên ba khía cạnh:

1.2.3.1 Phạm vi về Dữ liệu và Ngữ cảnh áp dụng

- **Khả năng của hệ thống:** EduTwin được thiết kế với kiến trúc "Metadata-driven" (hướng siêu dữ liệu), cho phép áp dụng linh hoạt cho nhiều mô hình giáo dục khác nhau (từ niên chế, tín chỉ đến các khóa học ngắn hạn) thông qua cơ chế định nghĩa cấu trúc động.
- **Phạm vi dữ liệu thực nghiệm:** Trong khuôn khổ khóa luận này, nhóm nghiên cứu giới hạn việc thu thập dữ liệu và kiểm thử các thuật toán dự báo dựa trên "mô hình Giáo dục phổ thông (THPT)". Đây là mô hình tiêu chuẩn dùng để đánh giá tính chính xác của thuật toán Lazy Learning trước khi mở rộng sang các mô hình phức tạp khác.

1.2.3.2 Phạm vi về Chức năng hệ thống

Hệ thống EduTwin được phát triển các phân hệ cốt lõi:

- **Phân hệ Quản trị (Admin Portal):** Tập trung vào tính linh hoạt cấu trúc và quản lý hệ thống.
 - ◊ Quản lý Cấu trúc động (Dynamic Structure Management): Cho phép Admin tự định nghĩa mô hình phân cấp dữ liệu (Ví dụ: Năm học → Học kỳ → Môn học; hoặc Khóa học → Module).
 - ◊ Quản lý Dataset tham chiếu: Upload và quản lý bộ dữ liệu mẫu (Excel) phục vụ thuật toán dự đoán.
 - ◊ Cấu hình tham số mô hình AI: Điều chỉnh các hyperparameter (k cho KNN, bandwidth cho KR, tau cho LWLR) và theo dõi hiệu suất dự báo (MAE, RMSE, Accuracy).
 - ◊ Quản lý tài liệu cấu trúc: Upload tài liệu tham khảo (PDF, DOCX) để cung cấp ngữ cảnh cho AI phân tích.
- **Phân hệ Người dùng (Student Portal):** Tập trung vào trải nghiệm cá nhân hóa.
 - ◊ Dashboard Analytics & Dự báo kết quả (What-if Analysis): Hiển thị biểu đồ điểm số và dự đoán kết quả tương lai.
 - ◊ Nhập và quản lý điểm số: Người dùng có thể nhập điểm thực tế theo từng môn học và thời điểm.
 - ◊ AI Companion (Chatbot Mode): Chatbot tư vấn học tập có khả năng ghi nhớ lịch sử hội thoại và cá nhân hóa phản hồi dựa trên hồ sơ người dùng.
 - ◊ Learning Agent (Learning Mode): AI Agent sử dụng kiến trúc ReAct (Reasoning + Acting), có khả năng sử dụng công cụ (Calculator, Wikipedia, Python REPL, Document Search) để giải đáp câu hỏi học tập phức tạp.
 - ◊ Upload tài liệu học tập: Người dùng có thể tải lên tài liệu (PDF, DOCX, TXT) để AI tìm kiếm và trích xuất thông tin liên quan.

1.2.3.3 Phạm vi về Công nghệ

- **Thuật toán:** Tập trung sâu vào nhóm thuật toán Lazy Learning (KNN, KR, LWLR), kỹ thuật Phân cụm KMeans và Tuyển chọn mẫu hình thích ứng (Adaptive Prototype

Selection).

- **Nền tảng:**

- **Backend:** Python (FastAPI) cho hiệu năng xử lý bắt đồng bộ cao.
- **Frontend:** ReactJS (Vite) tối ưu trải nghiệm người dùng.
- **Database:** PostgreSQL với JSONB để lưu trữ cấu trúc động và Redis để caching dữ liệu Cluster Index phục vụ dự đoán thời gian thực.
- **LLM:** Google Gemini 2.5 Flash với LangChain framework, cơ chế concurrency control (Semaphore) và retry logic để phục vụ nhiều người dùng đồng thời.
- **Deployment:** Docker containerization để chuẩn hóa môi trường triển khai, AWS ECS Fargate cho môi trường production.

1.3 Phương pháp nghiên cứu

Đề tài áp dụng cách tiếp cận Nghiên cứu hành động (Action Research) kết hợp với quy trình phát triển phần mềm hiện đại.

- **Nghiên cứu lý thuyết chuyên sâu:**

- Tổng hợp và phân tích các bài báo khoa học về Adaptive Learning Systems và Student Modeling.
- Nghiên cứu nguyên lý hoạt động và ưu nhược điểm của các họ thuật toán Eager Learning so với Lazy Learning trong bối cảnh dữ liệu giáo dục thay đổi nhanh (high volatility).
- Nghiên cứu kiến trúc AI Agent hiện đại: ReAct (Reasoning + Acting) và kỹ thuật Self-Reflection để xây dựng trợ lý học tập thông minh.

- **Phân tích thiết kế hệ thống hướng đối tượng:**

- Sử dụng Unified Modeling Language (UML) để mô hình hóa toàn bộ quy trình nghiệp vụ và kiến trúc hệ thống.
- Thiết kế kiến trúc hướng dịch vụ (Service-Oriented Architecture (SOA)) để đảm bảo tính mở rộng và khả năng bảo trì.
- Áp dụng kiến trúc Metadata-driven cho phép cấu hình động cấu trúc dữ liệu giáo dục.

- **Thực nghiệm và Đánh giá:**

- ◊ *Thu thập dữ liệu:* Xây dựng bộ dữ liệu mẫu (Representative Dataset) từ 2,283 học sinh thực tế tại các trường THPT Việt Nam, bao gồm 9 môn học qua 6 thời điểm (HK1, HK2 cho lớp 10, 11, 12).
- ◊ *Triển khai Prototype:* Xây dựng phiên bản Minimum Viable Product (MVP) để kiểm chứng tính khả thi của giải pháp công nghệ.
- ◊ *Đánh giá định lượng thuật toán dự đoán:* Sử dụng các chỉ số thống kê:
 - * Mean Absolute Error (MAE) (Mean Absolute Error): Đo sai số tuyệt đối trung bình.
 - * Root Mean Squared Error (RMSE) (Root Mean Square Error): Đo sai số bình phương trung bình.
 - * Accuracy (Within Threshold): Tỷ lệ dự đoán nằm trong ngưỡng chấp nhận được.
- ◊ *Stress Testing và Performance Benchmarking:* So sánh hiệu năng giữa phương pháp Global Scan ($O(N)$) và Cluster Indexing ($O(N/K)$) trên các tập dữ liệu từ 1,000 đến 100,000 mẫu. Đo lường thời gian phản hồi (Response Time) và hệ số tăng tốc (Speedup Factor).
- ◊ *Kiểm thử khả năng mở rộng:* Đánh giá khả năng phục vụ đồng thời 40-50 người dùng với cơ chế concurrency control và async processing.

1.4 Tổng quan đề tài

1.4.1 Các nghiên cứu và giải pháp liên quan

Trong bối cảnh chuyển đổi số giáo dục đang diễn ra mạnh mẽ, nghiên cứu này kế thừa và phát triển dựa trên bốn trụ cột công nghệ chính: Mô hình Bản sao số (Digital Twin), Hệ thống học tập thích ứng (Adaptive Learning), Trí tuệ nhân tạo tạo sinh (Generative AI), và các kỹ thuật xử lý dữ liệu khuyết thiếu. Dưới đây là phân tích tổng quan về các nghiên cứu gần đây và cách tiếp cận giải quyết vấn đề của EduTwin.

1.4.1.1 Digital Twin trong Giáo dục: Từ mô phỏng môi trường đến bản sao người học

Khái niệm Digital Twin ban đầu được phát triển trong lĩnh vực công nghiệp, tuy nhiên những năm gần đây đã chứng kiến sự dịch chuyển mạnh mẽ của công nghệ này sang lĩnh vực giáo dục và đào tạo. Các nghiên cứu tiên phong trong năm 2024 và 2025 đã minh chứng tiềm năng to lớn của Digital Twin trong việc tạo ra các môi trường thực tế ảo an toàn và hiệu quả. Điển hình, [2] và [3] đã áp dụng thành công Digital Twin để hỗ trợ đào tạo trong ngành xây dựng và kiến trúc cảnh quan, cho phép người học tương tác với các kịch bản mô phỏng phức tạp. Tương tự, trong lĩnh vực y tế, [4] đã phát triển bản sao số của bệnh nhân để nâng cao năng lực chẩn đoán cho sinh viên y khoa mà không gây rủi ro trên người thật. Mở rộng hơn về mặt phương pháp luận, [5] và [6] đề xuất tích hợp Digital Twin vào các mô hình học tập truy vấn (inquiry learning) và E-learning để tăng cường tính tương tác.

Tuy nhiên, điểm hạn chế căn bản của đa số các nghiên cứu hiện hành là sự tập trung quá mức vào việc mô phỏng "đối tượng bên ngoài" (như máy móc, công trình, hoặc bệnh nhân) thay vì chính bản thân người học. Các hệ thống này thường vận hành như những mô phỏng tĩnh hoặc bán động, thiếu vắng sự kết nối dữ liệu sinh trắc và tâm lý thời gian thực của chủ thể học tập. Khắc phục khoảng trống này, EduTwin đề xuất một cách tiếp cận lấy người học làm trung tâm, trong đó Digital Twin không phải là môi trường học, mà là bản sao kỹ thuật số của chính học sinh. Bằng cách coi dữ liệu điểm số và hành vi tương tác là các tín hiệu đầu vào liên tục, EduTwin xây dựng một thực thể số sống động, có khả năng tiến hóa song song và phản ánh chính xác trạng thái năng lực hiện tại của người học.

1.4.1.2 Học tập thích ứng và Bài toán dự báo hiệu suất: Hướng tiếp cận Lazy Learning

Mục tiêu quan trọng nhất của giáo dục cá nhân hóa là khả năng thích ứng theo thời gian thực với nhu cầu của người học. Các hệ thống gia sư thông minh - Intelligent Tutoring Systems (ITS) hiện đại đang nỗ lực giải quyết bài toán này thông qua việc phân tích dữ liệu lớn để đưa ra phản hồi cá nhân hóa, như được chỉ ra trong các nghiên cứu của [7] và [8]. Một số hệ thống tiên tiến hơn, như Tutomat được đề cập bởi [9], đã bắt đầu tích hợp xử lý ngôn ngữ tự nhiên để cải thiện giao diện tương tác, hay các nỗ lực tối ưu hóa thời gian thực bằng điện toán đám mây của [10].

Để đạt được khả năng thích ứng, nền tảng cốt lõi nằm ở độ chính xác của các mô hình dự báo hiệu suất. Xu hướng chủ đạo hiện nay, theo tổng hợp của [11] và [12], là sử dụng

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

các kỹ thuật Eager Learning, đặc biệt là Học sâu (Deep Learning) để dự đoán kết quả học tập. Mặc dù đạt hiệu suất cao, cách tiếp cận Eager Learning (như LSTM, Neural Networks) bộc lộ nhược điểm khi áp dụng vào môi trường giáo dục phổ thông: tính "hộp đen" khó giải thích và chi phí huấn luyện lại (retraining cost) quá lớn khi dữ liệu thay đổi liên tục. Hơn nữa, với các chuỗi dữ liệu học tập ngắn và ngắt quãng, các mô hình phức tạp thường không phát huy được ưu thế so với các phương pháp đơn giản hơn nhưng mạnh về cấu trúc cục bộ.

Đối mặt với thách thức này, EduTwin lựa chọn hướng đi khác biệt với chiến lược Lazy Learning (KNN, KR, LWLR). Thay vì cố gắng khai quát hóa toàn bộ dữ liệu vào một mô hình tĩnh, hệ thống trì hoãn quá trình tính toán đến thời điểm dự báo. Cách tiếp cận này không chỉ loại bỏ hoàn toàn độ trễ huấn luyện (zero-latency adaptation) mà còn đảm bảo tính minh bạch, khi mọi dự báo đều có thể được giải thích thông qua việc tham chiếu đến các hồ sơ "láng giềng" tương đồng trong quá khứ.

1.4.1.3 Generative AI và LLM: Từ công cụ tạo sinh đến trợ lý ngữ cảnh

Sự bùng nổ của Generative AI đã mở ra một kỷ nguyên mới cho giáo dục thông minh. [13] và [14] nhận định rằng các Mô hình Ngôn ngữ lớn (LLM) có tiềm năng chuyển đổi vai trò từ công cụ hỗ trợ thụ động sang các tác nhân chủ động. Các ứng dụng thực tiễn đã được triển khai đa dạng, từ việc tạo đề thi tự động [15] đến hỗ trợ đào tạo chuyên sâu trong các ngành kỹ thuật [16].

Tuy nhiên, việc tích hợp LLM vào môi trường giáo dục đối mặt với hai rào cản lớn: hiện tượng "ảo giác" (hallucination) và rủi ro bảo mật thông tin cá nhân [17]. Một trợ lý ảo đơn thuần sử dụng dữ liệu huấn luyện đại trà sẽ thiếu sự thấu hiểu về ngữ cảnh cụ thể của từng học sinh, dẫn đến các lời khuyên chung chung hoặc thiếu chính xác.

EduTwin giải quyết các vấn đề này thông qua ba cơ chế chính:

- **Kiến trúc ReAct Agent (Reasoning + Acting):** Thay vì sử dụng Retrieval-Augmented Generation (RAG) truyền thống, EduTwin triển khai AI Agent theo mẫu thiết kế Re-Act, cho phép hệ thống thực hiện suy luận nhiều bước (multi-step reasoning). Agent có khả năng sử dụng bộ công cụ tích hợp bao gồm: Máy tính (Calculator), Tra cứu Wikipedia, Thực thi Python (Python REPL), và Tìm kiếm tài liệu người dùng (Document Search). Cơ chế Self-Reflection cho phép Agent tự đánh giá chất lượng câu trả lời trước khi phản hồi, giảm thiểu hiện tượng hallucination.
- **Ngữ cảnh từ Digital Twin:** Hệ thống buộc LLM phải sinh câu trả lời dựa trên dữ

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

liệu điểm số thực tế được truy xuất từ Digital Twin của học sinh, thay vì "sáng tác" thông tin. Điều này đảm bảo các nhận xét và lời khuyên luôn bám sát thực tế học lực của người dùng.

- **Cơ chế Personally Identifiable Information (PII) Redaction:** Lớp bảo mật trung gian tự động ẩn danh hóa các thông tin định danh cá nhân (họ tên, email, số điện thoại, địa chỉ) trước khi gửi đến LLM. Kỹ thuật này sử dụng hàm băm SHA-256 và các regex pattern để phát hiện và che giấu thông tin nhạy cảm, tạo ra môi trường tương tác vừa thông minh, vừa an toàn về mặt quyền riêng tư.

Sự kết hợp của ba cơ chế này giúp EduTwin vượt qua các giới hạn của chatbot truyền thống, cung cấp trải nghiệm hỗ trợ học tập mang tính cá nhân hóa sâu sắc mà vẫn đảm bảo độ chính xác và an toàn thông tin.

1.4.1.4 Xử lý dữ liệu khuyết thiếu trong hồ sơ học tập

Dữ liệu giáo dục trong thực tế hiếm khi hoàn hảo; tình trạng dữ liệu thưa (sparse data) và khuyết thiếu là vấn đề thường trực ảnh hưởng đến độ chính xác của các thuật toán. Các phương pháp truyền thống như thay thế bằng giá trị trung bình (mean imputation) đã được chứng minh là không hiệu quả trong việc bảo toàn cấu trúc dữ liệu phức tạp. Các nghiên cứu mới nhất từ [18] và [19] đã khẳng định ưu thế vượt trội của các kỹ thuật điền khuyết dựa trên máy học. Đặc biệt, [20] và [21] chỉ ra rằng phương pháp k-Nearest Neighbors Imputation (KNNI) giúp cải thiện đáng kể độ chính xác của các bài toán dự báo nhờ khả năng tận dụng thông tin từ các mẫu tương đồng.

Vận dụng các kết quả này, EduTwin thiết lập quy trình xử lý dữ liệu "Fill-then-Predict". Thay vì loại bỏ các bản ghi thiếu hoặc điền giá trị ngẫu nhiên, hệ thống sử dụng KNNI để tái tạo các điểm số bị khuyết dựa trên mối tương quan đa biến giữa các môn học. Chiến lược này giúp khôi phục một hồ sơ năng lực hoàn chỉnh nhất có thể trước khi đưa vào các mô hình dự báo xu hướng, đảm bảo tính ổn định và độ tin cậy của toàn bộ hệ thống ngay cả khi dữ liệu đầu vào không liên mạch.

1.4.2 Đóng góp mới của đề tài

EduTwin khắc phục các hạn chế của các hệ thống truyền thống và đóng góp vào lĩnh vực Công nghệ Giáo dục (EdTech) thông qua bốn điểm mới cốt lõi sau:

- **Kiến trúc Dự báo Thích ứng Hiệu năng cao (High-Performance Adaptive Prediction Architecture):** Đề tài hiện thực hóa quy trình xử lý dữ liệu toàn diện theo cơ chế "**Fill-then-Predict**", kết hợp giữa kỹ thuật điền khuyết **KNNI** và thuật toán học lười (**KNN**, **KR**, **LWLR**). Điểm đột phá nằm ở việc ứng dụng kỹ thuật **Cluster Indexing (Phân cụm đánh chỉ mục)** kết hợp **Adaptive Prototype Selection (Tuyển chọn mẫu hình thích ứng)** để nén không gian tìm kiếm từ $O(N)$ xuống $O(N/K)$. Giải pháp này đã được kiểm chứng thực nghiệm trên bộ dữ liệu 100,000 mẫu, đạt hệ số tăng tốc **10-17 lần** so với phương pháp quét toàn bộ (Global Scan), giải quyết triệt để bài toán độ trễ của Lazy Learning mà không cần hạ tầng GPU đắt đỏ.
- **Kiến trúc Hệ thống hướng Siêu dữ liệu (Metadata-Driven Architecture):** Khác biệt với các LMS truyền thống, EduTwin giới thiệu mô hình "Cấu trúc động" (Dynamic Structure). Đóng góp này nằm ở việc tách biệt hoàn toàn tầng "Logic xử lý nghiệp vụ" khỏi tầng "Định nghĩa dữ liệu" sử dụng công nghệ **PostgreSQL JSONB**. Điều này cho phép các cơ sở giáo dục tự do định nghĩa các mô hình đánh giá phức tạp (đa cấp độ, đa trọng số) mà hệ thống AI vẫn tự động thích nghi và xử lý chính xác thông qua cơ chế ánh xạ động.
- **Learning Agent với Kiến trúc ReAct và Self-Reflection:** Đề tài xây dựng AI Agent theo mẫu thiết kế **ReAct (Reasoning + Acting)**, vượt xa khả năng của chatbot truyền thống. Agent có khả năng thực hiện suy luận nhiều bước (multi-step reasoning), sử dụng bộ công cụ tích hợp (Calculator, Wikipedia, Python REPL, Document Search), và tự đánh giá chất lượng câu trả lời thông qua cơ chế **Self-Reflection** trước khi phản hồi. Điều này giúp giảm thiểu hiện tượng "ảo giác" (hallucination) và nâng cao độ tin cậy của các phản hồi AI trong ngữ cảnh giáo dục.
- **Cơ chế Cá nhân hóa Hybrid và Bảo mật Thông tin:** Đề tài phát triển phương pháp **Hybrid Personalization** kết hợp nhận diện từ khóa (Keyword Detection) với phân tích LLM, cho phép học sở thích người dùng một cách hiệu quả về chi phí API. Hệ thống tích hợp dữ liệu điểm số định lượng từ Digital Twin với hồ sơ hành vi định tính để cá nhân hóa phản hồi. Đặc biệt, module **PII Redaction** sử dụng hàm băm SHA-256 và regex pattern để tự động ẩn danh hóa thông tin định danh (họ tên, email, số điện thoại) trước khi gửi đến LLM, đảm bảo an toàn quyền riêng tư cho học sinh.

1.5 Cấu trúc Khoa luận tốt nghiệp

Khóa luận với đề tài “EduTwin” được trình bày bao gồm 5 chương. Nội dung tóm tắt từng chương được trình bày như sau:

- **Chương 1: Tổng quan về đề tài.** Tổng quan về đề tài. Đặt vấn đề, xác định bài toán, mục tiêu và phạm vi nghiên cứu, đồng thời nêu bật tính cấp thiết và đóng góp của đề tài.
- **Chương 2: Cơ sở lý thuyết.** Trình bày các định nghĩa hình thức về Digital Twin, cơ sở toán học của các thuật toán Lazy Learning (KNN, KR, LWLR), cơ chế Attention trong LLM, các kỹ thuật điền khuyết dữ liệu (Imputation) được sử dụng làm nền tảng xây dựng hệ thống và các công nghệ phát triển Modern Web App.
- **Chương 3: Phân tích và Thiết kế hệ thống.** Mô tả chi tiết quy trình nghiệp vụ, kiến trúc phần mềm và thiết kế cơ sở dữ liệu quan hệ.
- **Chương 4: Thực nghiệm và Đánh giá.** Trình bày quá trình xây dựng mã nguồn (Implementation), minh họa các chức năng chính và phân tích kết quả thực nghiệm trên bộ dữ liệu mẫu.
- **Chương 5: Kết luận và Hướng phát triển.** Tổng kết các kết quả đạt được, nhìn nhận thẳng thắn các hạn chế và đề xuất lộ trình nâng cấp sản phẩm trong tương lai.

Chương 2. CƠ SỞ LÝ THUYẾT

Chương này trình bày hệ thống cơ sở lý thuyết khoa học và các giải pháp công nghệ được áp dụng trong việc xây dựng EduTwin. Nội dung tập trung phân tích sâu về mô hình Digital Twin trong giáo dục, các thuật toán học máy Lazy Learning và kiến trúc hệ thống, đồng thời lý giải chi tiết các quyết định kỹ thuật dựa trên đặc thù của bài toán.

2.1 Digital Twin và Học tập Cá nhân hóa

Digital Twin trong Bối cảnh Giáo dục. Về bản chất, Digital Twin (Bản sao số) là sự biểu diễn ảo hóa (virtual representation) của một thực thể, quy trình hoặc hệ thống trong thế giới thực. Điểm cốt lõi phân biệt Digital Twin với các mô hình mô phỏng tĩnh (simulation) nằm ở kết nối dữ liệu liên tục: luồng dữ liệu từ thực thể vật lý được truyền tải thời gian thực để cập nhật bản sao số, đảm bảo bản sao này luôn phản ánh chính xác trạng thái hiện tại của thực thể. Ngược lại, những tri thức (insights) phân tích từ bản sao số sẽ được dùng để hỗ trợ ra quyết định hoặc tác động tối ưu hóa lại thực thể thực.

EduTwin vận dụng nguyên lý này để xây dựng "Bản sao số người học". Trong mô hình này, người học đóng vai trò là thực thể vật lý, còn điểm số và hành vi tương tác là các tín hiệu đầu vào (thay cho cảm biến Internet of Things (IoT) trong công nghiệp). Không giống như các hồ sơ dữ liệu truyền thống vốn chỉ lưu trữ thông tin tĩnh và rời rạc (học bạ, điểm số quá khứ), Digital Twin trong EduTwin đóng vai trò là một thực thể số song song, có khả năng phản ánh trạng thái thực của người học. Hệ thống này hoạt động dựa trên cơ chế vòng lặp phản hồi kín (Closed-loop feedback): dữ liệu từ người học (điểm số, tương tác) liên tục được cập nhật vào bản sao số; bản sao số sử dụng các mô hình tính toán để dự báo và đưa ra các kịch bản tối ưu; cuối cùng, các tác động sư phạm được áp dụng ngược lại cho người học thực.

Tóm lại, EduTwin vận dụng triệt để nguyên lý này để chuyển đổi từ mô hình quản lý hồ sơ tĩnh sang một hệ thống Bản sao số động, nơi dữ liệu được luân chuyển liên tục trong một vòng lặp. Cách tiếp cận này giúp đồng bộ hóa trạng thái năng lực của người học, tạo tiền đề khoa học vững chắc cho việc cá nhân hóa và tối ưu hóa lộ trình phát triển.

Học tập Thích ứng (Adaptive Learning). Học tập cá nhân hóa, hay còn gọi là học tập thích ứng (adaptive learning), là phương pháp giáo dục sử dụng thuật toán máy tính để điều chỉnh nội dung, phương pháp giảng dạy và lộ trình học dựa trên nhu cầu, trình độ và phong cách học tập riêng biệt của từng cá nhân. Thay vì áp dụng mô hình "one-size-fits-all", hệ

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

thông EduTwin phân tích dữ liệu lịch sử để xác định chính xác "Vùng phát triển gần" Zone of Proximal Development (ZPD) theo lý thuyết của Lev Vygotsky (1978).

Quy trình cá nhân hóa được thực hiện bằng cách so khớp hồ sơ năng lực hiện tại của học sinh với cơ sở dữ liệu, sử dụng phương pháp tham chiếu dựa trên đồng nghiệp (peer-based reference). Việc cá nhân hóa này được thực hiện thông qua việc liên tục so khớp hồ sơ năng lực của học sinh hiện tại với cơ sở dữ liệu lịch sử khổng lồ, từ đó tìm ra các lô trình thành công của những người học có đặc điểm tương đồng. Theo Brusilovsky và Millán (2007), học tập cá nhân hóa có thể cải thiện tỷ lệ hoàn thành khóa học lên đến 15–25%.

2.2 Lazy Learning

Trong bối cảnh dự án EduTwin, việc mô hình hóa năng lực và dự báo quỹ đạo học tập của học sinh đòi hỏi một phương pháp tiếp cận có khả năng thích ứng cao với sự biến động liên tục của dữ liệu. Thay vì sử dụng các mô hình tham số toàn cục (global parametric models) cố định, nghiên cứu này áp dụng chiến lược Học lười (Lazy Learning), cụ thể là Học dựa trên thể hiện Instance-based Learning (IBL).

Cơ sở lý thuyết của chiến lược này dựa trên giả định về tính trơn cục bộ (local smoothness): các đầu ra của hàm mục tiêu thay đổi một cách mượt mà theo các biến đầu vào, do đó các điểm dữ liệu nằm gần nhau trong không gian đặc trưng sẽ có giá trị đích tương đồng.

Khác với các phương pháp "học hăng hái" (eager learning) cố gắng tìm kiếm một hàm tổng quát $f(x)$ trên toàn bộ không gian dữ liệu ngay trong quá trình huấn luyện, các phương pháp IBL chỉ lưu trữ dữ liệu huấn luyện và trì hoãn quá trình tính toán đến thời điểm dự báo. Khi một truy vấn (query point) xuất hiện, hệ thống sẽ xây dựng một hàm xấp xỉ cục bộ (local approximation) dựa trên tập hợp các láng giềng gần nhất.

Ba mô hình được sử dụng trong dự án này — KNN, KR, LWLR— đại diện cho sự phát triển tuần tự về độ phức tạp của hàm xấp xỉ cục bộ này.

KNN: Xấp xỉ Hằng số Cục bộ

Đây là dạng cơ bản nhất của học lười. Về mặt lý thuyết, KNN xấp xỉ hàm mục tiêu $f(x)$ bằng một hàm hằng số (constant function) trong vùng lân cận của điểm truy vấn. Đối với bài toán dự báo điểm số (hồi quy), giá trị dự đoán \hat{y} tại điểm truy vấn x_q được tính là trung bình cộng của k láng giềng gần nhất:

$$\hat{y}(x_q) = \frac{1}{k} \sum_{x_i \in \mathcal{N}_k(x_q)} y_i$$

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

Trong đó, $\mathcal{N}_k(x_q)$ là tập hợp k điểm dữ liệu trong tập huấn luyện có khoảng cách gần nhất với x_q theo một thước đo xác định.

KR: Xấp xỉ Trung bình có Trọng số

Khắc phục hạn chế về sự thay đổi rời rạc của KNN, KR đưa ra một hàm xấp xỉ mượt mà hơn. Mô hình này gán trọng số w_i cho mọi điểm dữ liệu dựa trên khoảng cách của nó tới điểm truy vấn thông qua một hàm nhân (Kernel function) K_h :

$$\hat{y} = \frac{\sum_{i=1}^n K_h(d(x, x_i))y_i}{\sum_{i=1}^n K_h(d(x, x_i))}$$

Tham số quan trọng nhất ở đây là độ rộng dải (bandwidth h). Nó quyết định bán kính ảnh hưởng hiệu quả của các điểm lân cận: h nhỏ giúp mô hình nhạy bén với dữ liệu cục bộ, trong khi h lớn giúp làm trơn nhiễu.

LWLR Xấp xỉ Tuyến tính Cục bộ

LWLR là bước tiến cao nhất trong nhóm mô hình này. Thay vì xấp xỉ bằng một giá trị trung bình, LWLR xấp xỉ hàm mục tiêu bằng một hàm tuyến tính $y = \theta^T x$ chỉ có giá trị cục bộ. Tại mỗi điểm truy vấn x_q , thuật toán tìm tham số θ tối ưu bằng cách cực tiểu hóa hàm mất mát bình phương sai số có trọng số:

$$\min_{\theta} \sum_i w^{(i)} (y^{(i)} - \theta^T x^{(i)})^2$$

Trong đó, trọng số $w^{(i)}$ thường được tính qua hàm Gaussian Kernel với tham số độ rộng dải τ :

$$w^{(i)} = \exp \left(-\frac{\|x^{(i)} - x_q\|^2}{2\tau^2} \right)$$

Phương pháp này cho phép mô hình nắm bắt được cả "xu hướng" (đạo hàm) của dữ liệu tại điểm xét, mang lại độ chính xác cao hơn cho các quỹ đạo học tập phức tạp.

Tổng hợp lại, việc kết hợp ba mô hình này cho phép hệ thống EduTwin khai thác triệt để thông tin từ dữ liệu quá khứ theo các mức độ chi tiết khác nhau: từ tìm kiếm tương đồng đơn giản (KNN), làm trơn dữ liệu (KR) đến mô hình hóa xu hướng cục bộ (LWLR).

2.3 Phân cụm K-Means và Tối ưu hóa Mẫu hình

Để giải quyết thách thức về khả năng mở rộng (scalability) và tốc độ phản hồi của các thuật toán Học lười (Lazy Learning) khi áp dụng trên quy mô lớn, nghiên cứu này đề xuất sử

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

dụng kỹ thuật **Phân cụm K-Means (K-Means Clustering)** kết hợp với chiến lược **Tuyển chọn Mẫu hình Thích ứng (Adaptive Prototype Selection)**. Đây là phương pháp tổ chức và đánh chỉ mục dữ liệu nhằm phân hoạch không gian dữ liệu thành các vùng đại diện, giúp cân bằng giữa độ chính xác dự báo và hiệu suất tính toán.

2.3.1 Thuật toán K-Means Clustering

K-Means là thuật toán phân cụm không giám sát, thực hiện ánh xạ không gian dữ liệu đầu vào $X \in \mathbb{R}^d$ vào một tập hợp rời rạc hữu hạn $C = \{\mu_1, \mu_2, \dots, \mu_K\}$, trong đó mỗi vector μ_k được gọi là **tâm cụm (centroid)** hoặc **mẫu hình đại diện (prototype)**.

Mục tiêu cốt lõi là phân hoạch không gian dữ liệu học sinh thành K vùng Voronoi riêng biệt (V_1, \dots, V_K), sao cho mọi điểm dữ liệu trong cùng một vùng đều được đại diện bởi tâm cụm μ_k tương ứng. Thuật toán K-Means thực hiện điều này bằng cách tối thiểu hóa hàm mục tiêu biến dạng (Distortion Function) J :

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x^{(n)} - \mu_k\|^2 \quad (2.1)$$

Trong đó:

- $x^{(n)}$: vector đặc trưng của học sinh thứ n .
- μ_k : tâm cụm (centroid) của cụm thứ k .
- r_{nk} : biến chỉ thị (indicator variable), nhận giá trị 1 nếu $x^{(n)}$ thuộc cụm k , và 0 nếu ngược lại.

Quá trình tối ưu hóa hàm J diễn ra theo cơ chế lặp EM (Expectation-Maximization):

Bước Gán (Assignment Step):

$$r_{nk} = \begin{cases} 1 & \text{nếu } k = \operatorname{argmin}_j \|x^{(n)} - \mu_j\|^2 \\ 0 & \text{trường hợp khác} \end{cases} \quad (2.2)$$

Bước Cập nhật (Update Step):

$$\mu_k = \frac{\sum_{n=1}^N r_{nk} x^{(n)}}{\sum_{n=1}^N r_{nk}} \quad (2.3)$$

2.3.2 Xác định số lượng cụm tối ưu

Số lượng cụm K được xác định động dựa trên kích thước tập dữ liệu N , với mục tiêu mỗi cụm chứa khoảng 3,000 mẫu để đảm bảo hiệu năng tối ưu cho thuật toán Lazy Learning:

$$K(N) = \begin{cases} 1 & \text{nếu } N < 3000 \\ \min\left(\left\lceil \frac{N}{3000} \right\rceil, 100\right) & \text{nếu } N \geq 3000 \end{cases} \quad (2.4)$$

Giới hạn trên $K_{max} = 100$ được áp dụng để tránh overhead quản lý quá nhiều cụm trong trường hợp dữ liệu cực lớn.

2.3.3 Kiến trúc hai tầng (2-Tier Architecture)

Việc áp dụng K-Means tạo tiền đề cho kiến trúc lai ghép 2 tầng được thiết kế như sau:

Tầng 1: Phân cụm và Đánh chỉ mục Toàn cục (Global Clustering/Indexing).

Hệ thống sử dụng các tâm cụm $\{\mu_1, \mu_2, \dots, \mu_K\}$ như một cơ chế "đánh chỉ mục". Thay vì tìm kiếm trên toàn bộ không gian N mẫu dữ liệu, hệ thống chỉ cần xác định cụm gần nhất với query. Tất cả các mẫu trong mỗi cụm được lưu trữ và sắp xếp theo khoảng cách đến tâm cụm (gần nhất đầu tiên).

Tầng 2: Tuyển chọn Mẫu hình Thích ứng (Adaptive Prototype Selection).

Tại thời điểm dự đoán, hệ thống thực hiện tuyển chọn động T mẫu ($T = 3000$) từ cụm được gán. Các mẫu ưu tiên là những điểm dữ liệu thực tế nằm gần tâm cụm μ_k nhất:

$$P_{k^*} = \{x_1, x_2, \dots, x_T\} \text{ với } \|x_i - \mu_{k^*}\| \leq \|x_{i+1} - \mu_{k^*}\| \quad (2.5)$$

Nếu cụm C_{k^*} có ít hơn T mẫu, hệ thống tự động gộp thêm mẫu từ các cụm lân cận (Neighbor Merging):

$$P_{k^*} = C_{k^*} \cup \bigcup_{j \in \mathcal{N}(k^*)} C_j \quad \text{cho đến khi } |P_{k^*}| \geq T \quad (2.6)$$

Trong đó $\mathcal{N}(k^*)$ là tập các cụm lân cận được sắp xếp theo khoảng cách centroid.

2.3.4 Quy trình suy diễn (Inference Pipeline)

Tại thời điểm dự báo (runtime), quy trình suy diễn được thực hiện qua 3 bước:

1. **Định vị (Localization):** Xác định cụm k^* mà học sinh mới x_{query} thuộc về bằng cách so khớp với bộ tâm cụm $\{\mu_1, \dots, \mu_K\}$. Độ phức tạp: $O(K)$.
2. **Tuyển chọn Mẫu hình (Prototype Selection):** Thu thập $T = 3000$ mẫu từ cụm k^* . Nếu cụm có nhiều hơn T mẫu, chọn T mẫu gần tâm nhất. Nếu ít hơn T mẫu, gộp từ các cụm lân cận.
3. **Hồi quy Cục bộ (Local Regression):** Áp dụng thuật toán học lười (KNN, KR hoặc LWLR) trên tập mẫu P_{k^*} . Độ phức tạp: $O(T)$ thay vì $O(N)$.

2.3.5 Kết quả thực nghiệm

Phương pháp tiếp cận này mang lại hiệu quả kép:

- **Tốc độ suy diễn:** Tiệm cận thời gian thực do $K + T \ll N$. Kết quả thực nghiệm trên bộ dữ liệu 100,000 mẫu cho thấy hệ số tăng tốc đạt **10-17 lần** so với phương pháp quét toàn bộ (Global Scan).
- **Khả năng giải thích:** Cao, dựa trên việc tham chiếu đến nhóm mẫu hình cụ thể có đặc điểm tương đồng với học sinh đang dự đoán.

2.4 Kiến trúc Transformer và Cơ chế Attention.

Nền tảng của các Mô hình Ngôn ngữ Lớn như Chat-GPT và Gemini là kiến trúc Transformer, được giới thiệu bởi Vaswani et al. (2017) trong nghiên cứu "Attention Is All You Need". Điểm đột phá của kiến trúc này nằm ở cơ chế Tự chú ý (Self-Attention), cho phép mô hình "nhìn" đồng thời toàn bộ chuỗi đầu vào thay vì xử lý tuần tự như các mạng hồi quy truyền thống.

Cơ chế Attention tính toán mức độ liên quan giữa các token trong chuỗi thông qua ba ma trận Query (Q), Key (K) và Value (V):

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V$$

Trong đó d_k là chiều của vector Key, đóng vai trò hệ số chuẩn hóa để ổn định gradient. Nhờ cơ chế này, khi học sinh hỏi về "môn Toán", mô hình có thể tự động liên kết với các thông tin liên quan như điểm số, chương trình học, và lịch sử tương tác trước đó trong ngữ cảnh.

Quản lý Bộ nhớ Hội thoại (Conversation Memory Management). Một thách thức quan trọng khi triển khai chatbot là giới hạn về độ dài ngữ cảnh (context window) của các mô hình LLM. Ví dụ, GPT-4 Turbo có giới hạn khoảng 128.000 token, trong khi Gemini Pro hỗ trợ đến 1 triệu token. Tuy nhiên, chi phí Application Programming Interface (API) tăng tuyến tính theo số token sử dụng.

EduTwin áp dụng chiến lược Cửa sổ Trượt (Sliding Window), chỉ giữ lại 20 tin nhắn gần nhất trong ngữ cảnh để cân bằng giữa chi phí API và chất lượng phản hồi. Cách tiếp cận này giúp cân bằng giữa chi phí vận hành và chất lượng phản hồi, đồng thời đảm bảo tính liên tục của cuộc trò chuyện xuyên suốt phiên làm việc.

2.5 Xử lý Dữ liệu và Kỹ thuật Imputation

Trong thực tế vận hành, dữ liệu giáo dục thường xuyên đối mặt với vấn đề khuyết thiếu (sparse data). Nghiên cứu đề xuất sử dụng kỹ thuật KNNI để xử lý vấn đề này.

KNN Imputation là phương pháp điền khuyết đa biến, dựa trên giả định rằng các điểm dữ liệu gần nhau trong không gian đặc trưng sẽ có các giá trị thành phần tương tự nhau. Về mặt lý thuyết, giá trị bị khuyết x_j của một điểm dữ liệu x sẽ được ước lượng dựa trên thông tin từ k láng giềng gần nhất tìm thấy trong tập huấn luyện:

$$\hat{x}_j = \frac{\sum_{y \in \mathcal{N}_k(x)} w(x, y) \cdot y_j}{\sum_{y \in \mathcal{N}_k(x)} w(x, y)}$$

Trong đó trọng số $w(x, y)$ thường được chọn tỉ lệ nghịch với khoảng cách Euclidean chuẩn $d(x, y)$:

$$d(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2}$$

Việc áp dụng KNN Imputation giúp tái tạo một vector đầu vào hoàn chỉnh, đảm bảo tính ổn định cho các thuật toán dự báo phức tạp ở giai đoạn sau (như LWLR) mà không làm méo mó cấu trúc tương quan giữa các môn học.

2.6 Kiến trúc AI Agent và Mẫu thiết kế ReAct

AI Agent là hệ thống tự động có khả năng nhận diện môi trường, đưa ra quyết định và thực hiện hành động để đạt mục tiêu. Khác với chatbot truyền thống chỉ trả lời câu hỏi, AI Agent có thể sử dụng công cụ (Tool Use) và thực hiện suy luận nhiều bước.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

Mẫu thiết kế ReAct (**Reasoning + Acting**) được giới thiệu bởi Yao et al. (2022) [22], kết hợp suy luận ngôn ngữ với hành động trong một vòng lặp:

1. **Thought:** Agent phân tích vấn đề và lập kế hoạch
2. **Action:** Chọn và thực thi công cụ phù hợp
3. **Observation:** Nhận kết quả từ công cụ
4. Lặp lại cho đến khi đủ thông tin để trả lời

EduTwin mở rộng ReAct với cơ chế **Self-Reflection**, cho phép Agent tự đánh giá chất lượng câu trả lời (đủ thông tin? chính xác? cần tìm thêm?) trước khi phản hồi, giảm thiểu hiện tượng hallucination.

2.7 Các chỉ số Đánh giá Hiệu năng Mô hình

Các chỉ số đánh giá là công cụ quan trọng giúp đo lường, so sánh và xác định mức độ hiệu quả của các mô hình học máy. Việc lựa chọn chỉ số phù hợp không chỉ phản ánh đúng bản chất bài toán mà còn đảm bảo tính khách quan trong việc đánh giá hiệu suất dự đoán. Trong nghiên cứu này, tùy theo loại mô hình và đầu ra (liên tục hoặc phân loại), các chỉ số được sử dụng bao gồm:

2.7.1 Sai số tuyệt đối trung bình (MAE - Mean Absolute Error)

Công thức tính:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Sai số tuyệt đối trung bình đo lường khoảng cách tuyệt đối trung bình giữa giá trị thực tế và giá trị dự đoán, phản ánh mức độ sai lệch trung bình mà mô hình tạo ra. Chỉ số này có ưu điểm dễ hiểu, đơn giản trong tính toán và ít bị ảnh hưởng bởi các giá trị ngoại lai so với các chỉ số khác. Về mặt diễn giải, MAE càng nhỏ chứng tỏ mô hình dự đoán càng chính xác và ổn định.

2.7.2 Căn bậc hai sai số bình phương trung bình (RMSE - Root Mean Squared Error)

Công thức tính:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Căn bậc hai sai số bình phương trung bình có ý nghĩa tương tự như MAE nhưng nhán mạnh nhiều hơn vào các sai số lớn thông qua việc bình phương độ lệch giữa giá trị thực tế và giá trị dự đoán. RMSE đặc biệt nhạy cảm với các giá trị ngoại lai, do đó thường được sử dụng trong các bài toán mà sai số lớn cần được kiểm soát chặt chẽ. Trong quá trình đánh giá mô hình, RMSE càng nhỏ đồng nghĩa với hiệu suất dự đoán của mô hình càng cao và mức độ rủi ro sai số lớn càng thấp.

2.7.3 Hệ số xác định (R^2 - R-squared)

Công thức:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Hệ số xác định đo lường mức độ mà mô hình có thể giải thích phuơng sai trong biến mục tiêu, với \bar{y} là giá trị trung bình của biến cần dự đoán. Chỉ số này cho biết tỷ lệ biến thiên của dữ liệu đầu ra được mô hình lý giải thành công. R^2 có giá trị nằm trong khoảng từ 0 đến 1, trong đó giá trị càng tiến gần 1 chứng tỏ mô hình càng phù hợp, có khả năng giải thích tốt sự biến động của dữ liệu thực tế. Như vậy, R^2 càng cao đồng nghĩa với việc mô hình có chất lượng dự đoán càng tốt.

2.8 Trực quan hóa Dữ liệu Giáo dục.

Việc lựa chọn loại biểu đồ phù hợp đóng vai trò then chốt trong việc truyền tải thông tin học tập một cách hiệu quả. Hệ thống EduTwin áp dụng ba loại biểu đồ chính, mỗi loại phục vụ một mục đích phân tích riêng biệt:

Biểu đồ Radar (Radar Chart) được sử dụng để so sánh năng lực đa chiều của học sinh trên nhiều môn học đồng thời. Loại biểu đồ này cho phép nhận diện nhanh các điểm mạnh và điểm yếu tương đối, phù hợp với bài toán phân tích hồ sơ năng lực toàn diện. Trong EduTwin, mỗi trục của biểu đồ Radar đại diện cho một môn học, và diện tích vùng phủ phản ánh mức độ cân bằng của năng lực.

Biểu đồ Đường (Line Chart) được tối ưu cho việc thể hiện xu hướng biến đổi theo thời gian. Đây là công cụ chủ đạo trong việc trực quan hóa quỹ đạo học tập (learning trajectory),

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

cho phép học sinh và giáo viên theo dõi tiến bộ qua các kỳ học hoặc năm học. Khả năng nêu các điểm dữ liệu liên tiếp giúp người xem dễ dàng nhận ra các mẫu xu hướng tăng, giảm hoặc ổn định.

Biểu đồ Cột (Bar Chart) phù hợp cho việc so sánh trực tiếp giữa các danh mục rời rạc, ví dụ như so sánh điểm số giữa các môn trong cùng một kỳ học, hoặc so sánh hiệu suất giữa các khối thi (Tự nhiên, Xã hội). Tính trực quan của biểu đồ cột giúp học sinh nhanh chóng xác định môn học cần cải thiện.

Việc kết hợp ba loại biểu đồ này tạo ra một hệ thống trực quan hóa đa tầng, cho phép học sinh tiếp cận dữ liệu học tập từ nhiều góc độ khác nhau: từ góc nhìn tổng thể (Radar), đến phân tích xu hướng (Line), và so sánh chi tiết (Bar).

2.9 Xác thực API với Session-based Authentication

Trong kiến trúc web truyền thống, Session-based authentication sử dụng cơ chế lưu trữ trạng thái phiên làm việc trên server. EduTwin áp dụng phương pháp này với HTTP-only cookies để đảm bảo an toàn.

Quy trình xác thực:

1. Người dùng đăng nhập với username/password.
2. Server xác thực credentials và tạo session ID duy nhất.
3. Session ID được lưu trong Redis (server-side) và gửi về client qua HTTP-only cookie.
4. Mỗi request tiếp theo, browser tự động gửi cookie kèm session ID.
5. Server tra cứu session trong Redis để xác thực người dùng.

Ưu điểm của HTTP-only cookies:

- Không thể truy cập từ JavaScript → miễn nhiễm với tấn công XSS.
- Session có thể bị thu hồi ngay lập tức (revoke) từ phía server.
- Không cần quản lý token refresh phức tạp.

Chương 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Chương này trình bày chi tiết quá trình phân tích yêu cầu và thiết kế hệ thống EduTwin. Nội dung bao gồm việc xác định bài toán, mô hình hóa quy trình nghiệp vụ, thiết kế kiến trúc tổng thể, cơ sở dữ liệu và các thuật toán cốt lõi.

3.1 Phát biểu bài toán

3.1.1 Bối cảnh

Trong bối cảnh giáo dục hiện đại, việc cá nhân hóa trải nghiệm học tập cho từng học sinh đang trở thành xu hướng tất yếu. Tuy nhiên, các hệ thống hỗ trợ giáo dục hiện tại (như SMAS, VnEdu, Google Classroom) chủ yếu tập trung vào việc **lưu trữ và hiển thị** kết quả học tập, thiếu khả năng **phân tích, dự đoán và tư vấn** cá nhân hóa.

3.1.2 Bài toán cần giải quyết

Mục tiêu chính là xây dựng hệ thống "Bản sao học tập kỹ thuật số" (Digital Twin for Education) có khả năng:

- Dự đoán kết quả học tập:** Sử dụng điểm số hiện tại để dự đoán điểm các học kỳ/khoa học tiếp theo với độ chính xác cao.
- Thích ứng cấu trúc động:** Hỗ trợ nhiều chương trình giáo dục khác nhau (THPT, Đại học, Trung tâm) với số lượng môn học và học kỳ tùy biến - không cố định như các mô hình Deep Learning truyền thống.
- Tư vấn thông minh:** Cung cấp Chatbot AI có khả năng phân tích tình hình học tập, đưa ra lời khuyên dựa trên dữ liệu thực tế và tăng cường cá nhân hóa thông qua việc học tập thông tin trong quá trình trò chuyện.
- Hiệu năng cao:** Đảm bảo thời gian phản hồi nhanh ($< 200ms$ cho dự đoán) ngay cả khi tập dữ liệu tham chiếu lớn.

3.1.3 Các thách thức kỹ thuật và Giải pháp

Để đạt được các mục tiêu trên, hệ thống cần giải quyết các thách thức kỹ thuật sau:

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Thách thức	Giải pháp đề xuất
Input/Output không cố định	Sử dụng Lazy Learning (KNN, LWLR, KR) thay vì Deep Learning cố định kiến trúc.
Tốc độ chậm với dataset lớn	Áp dụng Phân cụm (Cluster) và Tối ưu hóa mẫu hình (Prototype Optimization với K-Means Clustering).
Bảo mật thông tin cá nhân	Cơ chế PII Redaction và Mã hóa trường dữ liệu (Field Encryption).
Cấu trúc chương trình học đa Kiến trúc hướng siêu dữ liệu (Metadata-driven dạng	Architecture) với PostgreSQL JSONB.

Bảng 3.1. Các thách thức kỹ thuật và Giải pháp đề xuất

3.2 Quy trình thực hiện

Quy trình phát triển hệ thống EduTwin được chia thành 4 giai đoạn chính, tuân thủ mô hình phát triển phần mềm linh hoạt (Agile) kết hợp với quy trình nghiên cứu khoa học.



Hình 3.1. Tổng quan quy trình thực hiện dự án

Chi tiết từng giai đoạn:

- **Giai đoạn 1: Phân tích yêu cầu.**

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Thu thập yêu cầu từ khảo sát thực tế và nghiên cứu các hệ thống hiện có để xác định danh sách yêu cầu chức năng và phi chức năng.

- **Giai đoạn 2: Thiết kế hệ thống.**

Xây dựng kiến trúc tổng thể, thiết kế cơ sở dữ liệu (Entity Relationship Diagram (ERD)), thuật toán Machine Learning (ML) (KNN, KR, LWLR, K-Means Clustering), kiến trúc AI Agent (ReAct) và thiết kế giao diện (UI/UX).

- **Giai đoạn 3: Triển khai.**

Xây dựng Backend (FastAPI), Frontend (ReactJS), tích hợp ML Pipeline (Scikit-learn) với Cluster Indexing, tích hợp LLM (Gemini) với Learning Agent và lớp bảo mật PII Redaction.

- **Giai đoạn 4: Kiểm thử và đánh giá.**

Thực hiện Unit Test, Integration Test, Stress Test hiệu năng (so sánh Global Scan vs Cluster Indexing), đo lường độ trễ (Latency) và đánh giá độ chính xác mô hình (RMSE, MAE, Accuracy).

3.3 Phân tích yêu cầu hệ thống

3.3.1 Yêu cầu chức năng

A. Nhóm chức năng cho Học sinh (Student Actor)

Học sinh là đối tượng người dùng chính của hệ thống EduTwin, với nhu cầu học tập cá nhân hóa và theo dõi tiến độ học tập của bản thân. Nhóm chức năng này được thiết kế để hỗ trợ học sinh trong suốt hành trình học tập, từ việc theo dõi điểm số, dự đoán kết quả học tập, đến việc tương tác với AI để nhận được sự hỗ trợ học tập phù hợp với phong cách học tập cá nhân. Hệ thống cung cấp hai chế độ tương tác AI chính: **Chatbot AI** cho các câu hỏi và tư vấn học tập tổng quát, và **Learning Agent** sử dụng kiến trúc ReAct để hỗ trợ học tập chuyên sâu với khả năng suy luận đa bước. Bên cạnh đó, tính năng dự đoán điểm sử dụng các thuật toán Lazy Learning (KNN, KR, LWLR) giúp học sinh có cái nhìn trước về kết quả học tập để điều chỉnh kế hoạch học tập kịp thời.

Các chức năng trong Bảng 1 được tích hợp chặt chẽ với nhau để tạo thành một hệ sinh thái học tập hoàn chỉnh. Dữ liệu điểm số của học sinh (F1.6) không chỉ được sử dụng để hiển thị trên Dashboard (F1.2) mà còn làm đầu vào cho mô hình dự đoán (F1.5) và cung cấp ngữ cảnh cho việc cá nhân hóa phản hồi của AI Agent (F1.3, F1.4). Tài liệu học tập

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

ID	Chức năng	Mô tả	Endpoint API
F1.1	Đăng ký/Đăng nhập	Xác thực với Bcrypt.	POST /auth/login, /auth/register
F1.2	Xem Dashboard	Tổng quan điểm số, dự đoán.	GET /custom- model/user- scores/{id}
F1.3	Chat với AI	Trò chuyện cá nhân hóa.	POST /chatbot, /chatbot/stream
F1.4	Learning Agent	Học tập với ReAct Agent.	POST /learning/chat
F1.5	Dự đoán điểm	Dự đoán điểm tương lai.	GET /custom- model/predict/{id}
F1.6	Cập nhật điểm	Nhập điểm thực tế.	POST /custom- model/user-scores
F1.7	Upload tài liệu	Tải tài liệu học tập.	POST /learning/upload- document
F1.8	Quản lý session	Tạo/xóa phiên chat.	/chatbot/session, /chatbot/sessions

Bảng 3.2. Danh sách chức năng cho Học sinh

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

được upload (F1.7) sẽ được xử lý và lưu trữ trong vector database để Learning Agent có thể truy xuất thông tin phù hợp khi trả lời câu hỏi của học sinh.

B. Nhóm chức năng cho Quản trị viên (Admin Actor)

Quản trị viên đóng vai trò quan trọng trong việc cấu hình và tối ưu hóa hệ thống EduTwin cho từng cơ sở giáo dục cụ thể. Nhóm chức năng này cho phép admin thiết lập cấu trúc môn học (Teaching Structure), quản lý dữ liệu tham chiếu, điều chỉnh tham số mô hình Machine Learning, và đánh giá hiệu suất của các thuật toán dự đoán khác nhau. Khái niệm **Teaching Structure** là trung tâm của hệ thống, định nghĩa cấu trúc điểm số của một môn học cụ thể (ví dụ: Toán học gồm 4 bài kiểm tra, 1 thi giữa kỳ, 1 thi cuối kỳ với trọng số khác nhau). Mỗi structure có thể được cấu hình độc lập với dataset riêng, tài liệu tham khảo riêng, và tham số mô hình Machine Learning riêng. Hệ thống hỗ trợ đa structure nhưng chỉ có một structure được kích hoạt (active) tại một thời điểm.

Quy trình làm việc điển hình của Admin bao gồm: (1) Tạo Teaching Structure mới với định nghĩa các thành phần điểm số và trọng số (F2.2); (2) Upload dataset tham chiếu chứa dữ liệu điểm số lịch sử của học sinh từ các năm trước (F2.4); (3) Upload tài liệu tham khảo như giáo trình, đề cương môn học (F2.5); (4) Sử dụng chức năng đánh giá mô hình để so sánh RMSE, MAE, và độ chính xác của ba thuật toán KNN, Kernel Regression, và LWLR trên dataset (F2.8); (5) Dựa trên kết quả đánh giá, chọn thuật toán tối ưu (F2.7) và tinh chỉnh các tham số như K (số láng giềng), bandwidth (độ rộng kernel), hoặc tau (tham số trọng số cục bộ) (F2.6); và cuối cùng (6) Kích hoạt structure để áp dụng cho toàn bộ học sinh (F2.3).

C. Nhóm chức năng AI Core (System Actor)

AI Core là thành phần nền tảng cung cấp các khả năng trí tuệ nhân tạo và machine learning cho toàn bộ hệ thống EduTwin. Nhóm chức năng này hoạt động ở tầng backend, xử lý các tác vụ phức tạp từ dự đoán điểm số, cá nhân hóa nội dung học tập, đến bảo vệ dữ liệu cá nhân của học sinh. Kiến trúc AI Core được thiết kế với ba trụ cột chính: (1) **Machine Learning Pipeline** sử dụng các thuật toán Lazy Learning với tối ưu hóa hiệu suất thông qua cluster indexing cho dataset lớn; (2) **LLM-based Personalization** kết hợp kỹ thuật trích xuất từ khóa và Large Language Model để tạo trải nghiệm học tập cá nhân hóa; và (3) **ReAct Agent Framework** cho phép AI thực hiện suy luận đa bước và sử dụng các công cụ (tools) để trả lời câu hỏi phức tạp.

Các chức năng trong AI Core được thiết kế để giải quyết các thách thức thực tiễn trong môi trường giáo dục. Chức năng KNN Imputation (F3.1) giải quyết vấn đề dữ liệu bị khuyết

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

ID	Chức năng	Mô tả	Endpoint API
F2.1	Xem tất cả Struc-Danh sách cấu trúc.		GET /custom-model/teaching-structures
F2.2	Câu hình Structure Tạo cấu trúc mới.		POST /custom-model/teaching-structure
F2.3	Kích hoạt Structure Chọn structure active.		POST /custom-model/activate/{id}
F2.4	Upload Dataset Tải file Excel tham chiếu.		POST /custom-model/upload-dataset/{id}
F2.5	Upload tài liệu Tải PDF/DOCX tham khảo. Structure		POST /developer/structure-document
F2.6	Tham số ML Điều chỉnh K, Bandwidth, Tau.	PUT	/developer/model-parameters
F2.7	Chọn model ML Chọn KNN/KR/LWLR.	POST	/developer/model-select
F2.8	Đánh giá model So sánh RMSE/MAE/Accuracy.	POST	/custom-model/evaluate-models

Bảng 3.3. Danh sách chức năng cho Admin

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

ID	Chức năng	Mô tả chi tiết
F3.1	KNN Imputation	Điền khuyết dữ liệu với khoảng cách NaN-aware.
F3.2	Lazy Learning Prediction	Dự đoán với KNN/KR/LWLR.
F3.3	Cluster Indexing	Phân cụm K-Means tối ưu tốc độ khi $N > 3000$.
F3.4	PII Redaction	Ẩn danh hóa (Tên, SĐT, Email) trước khi gửi LLM.
F3.5	ReAct Agent	Suy luận nhiều bước với 4 tools.
F3.6	Hybrid Personalization	Học sở thích qua Keyword + LLM.
F3.7	Document Processing	Xử lý PDF/DOCX, trích xuất nội dung.

Bảng 3.4. Danh sách chức năng của AI Core

do học sinh chưa hoàn thành một số bài kiểm tra, sử dụng khoảng cách NaN-aware để tìm các học sinh tương tự và điền giá trị hợp lý. Cluster Indexing (F3.3) được kích hoạt tự động khi số lượng mẫu trong dataset vượt quá 3000, sử dụng thuật toán K-Means để phân nhóm dữ liệu và giảm không gian tìm kiếm, qua đó cải thiện đáng kể tốc độ dự đoán mà vẫn duy trì độ chính xác cao. Về khía cạnh bảo mật và quyền riêng tư, chức năng PII Redaction (F3.4) đảm bảo rằng các thông tin nhận dạng cá nhân như tên, số điện thoại, và email được ẩn danh hóa trước khi dữ liệu được gửi đến LLM bên thứ ba (Google Gemini). Điều này tuân thủ các quy định bảo vệ dữ liệu cá nhân và giảm thiểu rủi ro rò rỉ thông tin nhạy cảm. Hybrid Personalization (F3.6) là một đóng góp quan trọng của hệ thống, kết hợp hai phương pháp bổ trợ: trích xuất từ khóa từ các câu hỏi và phản hồi của học sinh để xác định chủ đề quan tâm, và sử dụng LLM để phân tích phong cách học tập dựa trên lịch sử tương tác. Thông tin này được lưu trữ trong bảng `user_preferences` và được sử dụng để tùy chỉnh prompt gửi đến LLM, tạo ra các phản hồi phù hợp hơn với nhu cầu và phong cách học tập của từng học sinh cụ thể.

3.3.2 Yêu cầu phi chức năng

A. Hiệu năng (Performance)

- **API Response Time (Dự đoán):** $< 200ms$. Giải pháp: Sử dụng Lazy Learning kết hợp K-Means Clustering và Redis Cache.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- **API Response Time (Chat):** < 3s. Giải pháp: Tối ưu hóa ngữ cảnh (Sliding Window 20 messages) và Concurrency Control.
- **Cache Strategy:** Sử dụng chiến lược Multi-tier caching với Redis:
 - ◊ Prediction Cache: Time To Live (TTL) = 1 giờ (3600s)
 - ◊ Evaluation Cache: TTL = 2 giờ (7200s)
 - ◊ Cluster Index Cache: TTL = 24 giờ (86400s)

Cache key tính bằng SHA256 hash của tham số đầu vào.

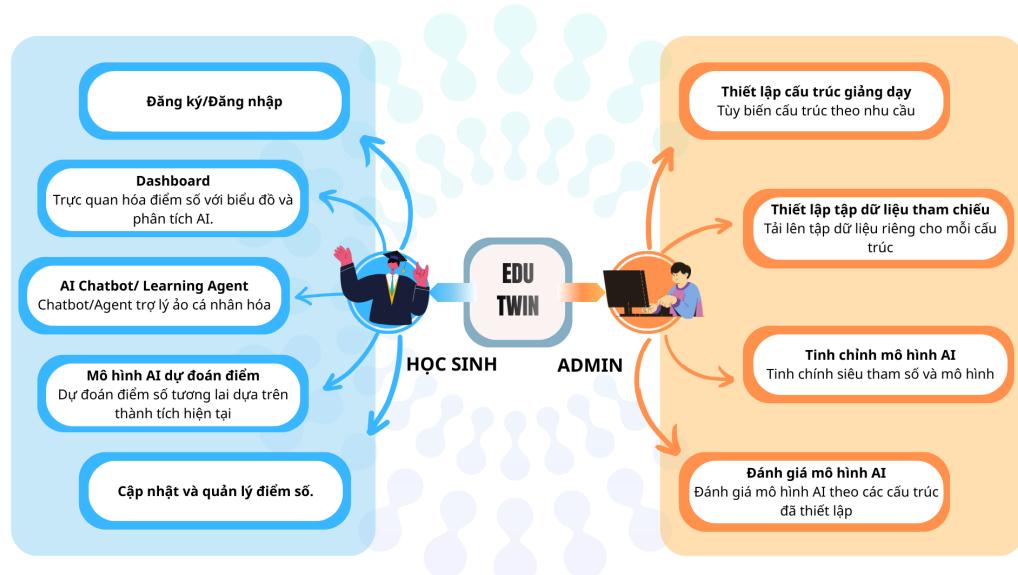
B. Bảo mật (Security)

- **Mã hóa mật khẩu:** Sử dụng Bcrypt (thư viện Passlib) với độ phức tạp mặc định 12 rounds.
- **Xác thực:** Sử dụng Session-based authentication với HTTP-only cookies, đảm bảo an toàn trước tấn công XSS.
- **Mã hóa PII:** Sử dụng Fernet symmetric encryption (AES-128) thông qua thư viện Cryptography. Các trường được mã hóa bao gồm: Email, Số điện thoại, Địa chỉ.
- **PII Redaction:** Lớp trung gian ẩn danh hóa thông tin cá nhân trước khi gửi đến LLM:
 - ◊ Họ tên → SHA-256 hash (ví dụ: USER_A1B2C3D4)
 - ◊ Email → Mask (ví dụ: n***n@g***l.com)
 - ◊ Số điện thoại → Mask (ví dụ: PHONE_XXXX5678)
 - ◊ Địa chỉ → Chỉ giữ Quận/Thành phố

3.4 Mô hình hóa quy trình nghiệp vụ

3.4.1 Biểu đồ Use Case

Biểu đồ Use Case dưới đây mô tả sự tương tác giữa các tác nhân chính (Học viên và Quản trị viên) với các chức năng cốt lõi của hệ thống EduTwin.



Hình 3.2. Biểu đồ Usecase của EduTwin

3.4.2 Quy trình Đăng nhập và Xác thực

1. Người dùng nhập username và password.
2. Hệ thống truy vấn database, kiểm tra username tồn tại.
3. Nếu tồn tại, so khớp password với hash đã lưu (Bcrypt).
4. Nếu khớp, tạo Session ID và lưu vào Redis.
5. Trả về Session ID qua HTTP-only cookie (không thể truy cập từ JavaScript).
6. Các request tiếp theo, browser tự động gửi cookie kèm Session ID.
7. Server tra cứu session trong Redis để xác thực người dùng.

3.4.3 Quy trình Dự đoán Điểm số

Đây là quy trình nghiệp vụ cốt lõi, được kích hoạt tự động khi người dùng cập nhật điểm:

1. Người dùng nhập hoặc cập nhật điểm số thực tế.
2. Hệ thống lưu điểm vào database và kiểm tra cấu trúc hiện hành.
3. Nếu có điểm thiếu, gọi **KNN Imputer** để điền khuyết.
4. Kiểm tra Redis Cache:
 - Nếu Hit: Sử dụng Cluster Index đã cache.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- Nếu Miss: Build Cluster Index mới (K-Means) và cache.

5. Gán người dùng vào Cluster gần nhất ($O(K)$).
6. Tuyển chọn ~3000 Prototypes từ cluster (hoặc gộp từ clusters lân cận).
7. Chạy thuật toán Lazy Learning (KNN/KR/LWLR) trên tập mẫu hình.
8. Lưu kết quả dự đoán vào database, cập nhật Cache và trả về client.

3.4.4 Quy trình Chat với AI (Chatbot Mode)

1. Người dùng gửi tin nhắn qua giao diện Chat.
2. Backend nhận request, load 20 tin nhắn gần nhất từ database (Sliding Window).
3. Xây dựng context: điểm số + sở thích đã học + lịch sử hội thoại.
4. Áp dụng **PII Redaction** trên toàn bộ context.
5. Gửi request đến Gemini API.
6. Nhận response và gửi về client qua WebSocket (Socket.IO).
7. Lưu message vào bảng `chat_messages`.
8. Chạy **Hybrid Personalization** để học sở thích người dùng:
 - Phân tích keyword (250+ từ khóa tiếng Việt).
 - Nếu đủ 8 meaningful messages → trigger LLM analysis.
 - Cập nhật preferences trong user profile.

3.4.5 Quy trình Learning Agent (Learning Mode)

1. Người dùng gửi câu hỏi học tập qua giao diện Learning.
2. Backend khởi tạo ReAct Agent với 4 tools: Calculator, Wikipedia, Python REPL, Document Search.
3. **Auto-search:** Agent tự động tìm kiếm trong tài liệu người dùng trước tiên.
4. Vòng lặp ReAct (tối đa 5 iterations):
 - **Thought:** LLM phân tích và lập kế hoạch.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- **Action:** Chọn và thực thi tool phù hợp.
- **Observation:** Nhận kết quả từ tool.
- Gửi trạng thái real-time về client qua WebSocket.

5. **Self-Reflection:** Agent tự đánh giá (đủ thông tin? chính xác?).
6. Nếu đủ → sinh Final Answer và trả về client.
7. Lưu conversation vào database.

3.4.6 Quy trình Thiết lập Cấu trúc Học tập

1. Admin truy cập trang Developer Tools.
2. Nhập tên cấu trúc (VD: "THPT Quốc Gia 2024").
3. Thêm danh sách môn học (subject_labels) và các mốc thời gian (time_point_labels).
4. Chọn thang điểm (scale_type: 0-10, 0-100, GPA).
5. Nhấn "Lưu" - hệ thống tạo record trong custom_teaching_structures.
6. Kích hoạt cấu trúc (is_active = true) - các cấu trúc khác tự động bị vô hiệu hóa.

3.4.7 Quy trình Upload Dataset và Đánh giá Model

1. Admin chọn cấu trúc và upload file Excel chứa dữ liệu tham chiếu.
2. Hệ thống parse file, kiểm tra header khớp với subject_labels × time_point_labels.
3. Nếu hợp lệ, lưu các record vào custom_dataset_samples.
4. Admin chọn Input Timepoints và Output Timepoints.
5. Admin điều chỉnh tham số: k (KNN), Bandwidth (KR), τ (LWLR).
6. Nhấn "Evaluate" - hệ thống đánh giá với Cross-validation hoặc 80/20 split.
7. Hiển thị kết quả: MAE, RMSE, Accuracy cho từng model.
8. Admin chọn model tốt nhất và kích hoạt pipeline.

3.4.8 Quy trình Upload Tài liệu Học tập (User)

1. Người dùng upload file PDF/DOCX/TXT qua giao diện Learning.
2. Backend xác định loại file và trích xuất text:
 - PDF: PyMuPDF hoặc pdfplumber.
 - DOCX: python-docx.
3. Lưu nội dung vào uploaded_documents trong user profile.
4. Khi Learning Agent hoạt động, **SearchUserDocuments** tool sẽ tìm kiếm keyword trong các tài liệu này.

3.4.9 Kiến trúc tổng thể

Hệ thống được thiết kế theo mô hình phân tầng (Layered Architecture):

- **Client Layer:** Ứng dụng Web SPA xây dựng bằng React 19 và Vite, kết nối real-time qua WebSocket (Socket.IO).
- **Backend Services:** FastAPI xử lý REST API và WebSocket. Các module: Auth, User, Chatbot, Learning Agent, Prediction.
- **Data Tier:** PostgreSQL (dữ liệu bền vững) và Redis (caching, session).
- **External Integration:** Kết nối với Google Gemini API qua lớp bảo mật (PII Redaction, Concurrency Control).

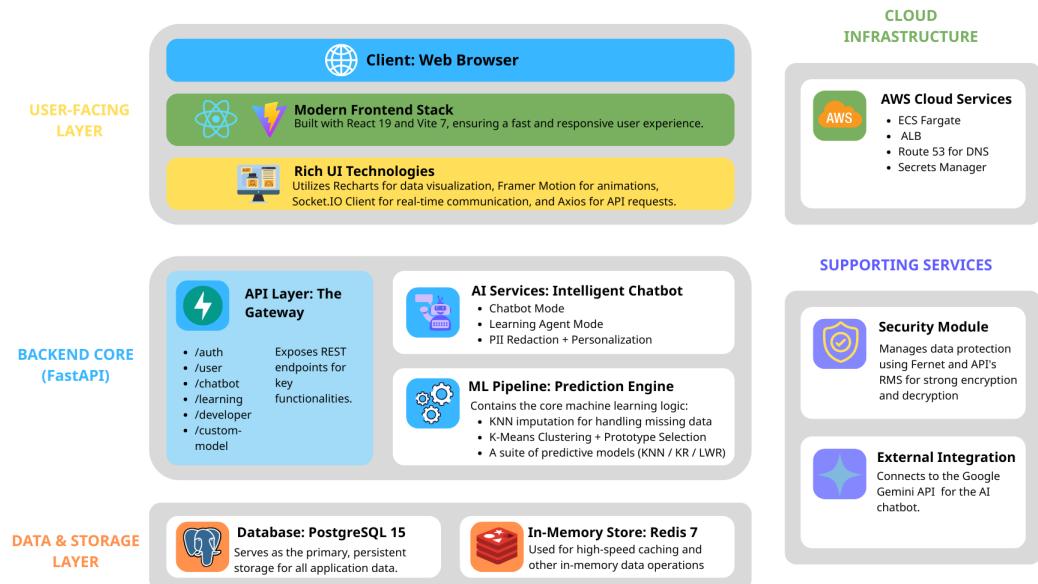
3.4.10 Kiến trúc Module AI

Module AI bao gồm hai pipeline xử lý chính:

A. Pipeline Dự đoán Điểm số (ML Prediction):

- **Preprocessor:** Xác thực dữ liệu theo cấu trúc động (JSONB).
- **Imputation Engine:** Xử lý dữ liệu khuyết bằng KNN Imputer.
- **Cluster Indexer:** Phân cụm K-Means và tuyển chọn mẫu hình thích ứng.
- **Predictor:** Thực thi KNN, KR, LWLR trên tập prototypes.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG



Hình 3.3. Kiến trúc tổng thể hệ thống EduTwin

B. Pipeline Tương tác AI:

- Chatbot Mode:** Gọi trực tiếp Gemini API với context (điểm số, lịch sử hội thoại, sở thích đã học). Áp dụng PII Redaction trước khi gửi.
- Learning Mode:** Sử dụng ReAct Agent với 4 tools:
 - Calculator (numexpr)
 - Wikipedia (tiếng Việt)
 - Python REPL (sandbox)
 - Document Search (tài liệu người dùng)

Kèm cơ chế Self-Reflection để tự đánh giá câu trả lời.

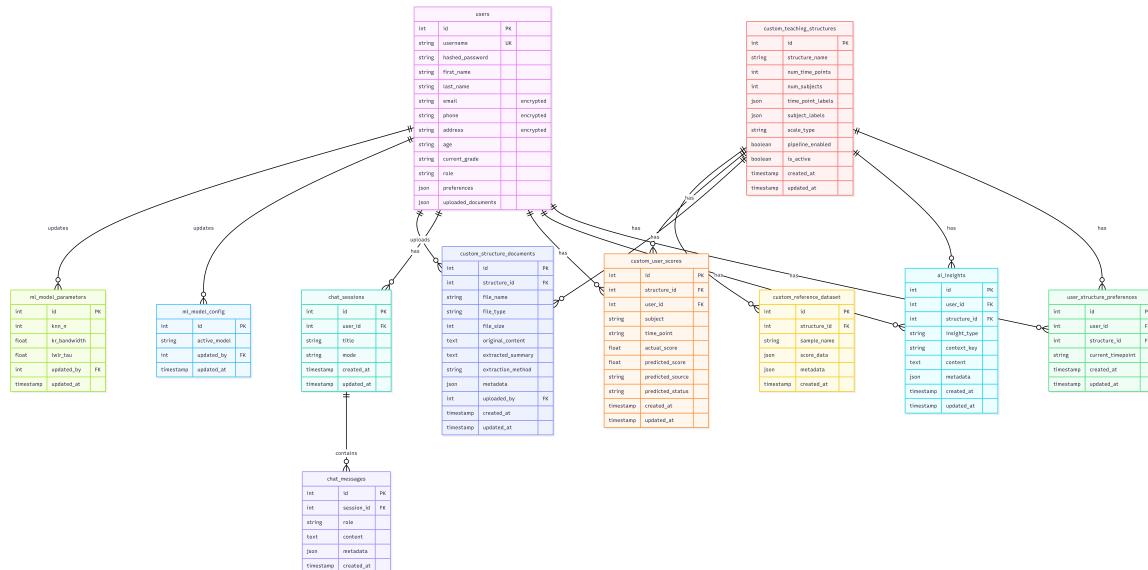
3.5 Thiết kế Cơ sở dữ liệu

Cơ sở dữ liệu của EduTwin được thiết kế dựa trên hệ quản trị cơ sở dữ liệu **PostgreSQL 15**, sử dụng kiến trúc lai ghép (Hybrid Model). Kiến trúc này kết hợp tính chât chẽ, toàn vẹn dữ liệu của mô hình quan hệ (Relational) cho thông tin người dùng, với tính linh hoạt của mô hình tài liệu (Document-based) thông qua kiểu dữ liệu **JSONB** để lưu trữ các cấu trúc học tập động.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.5.1 Thiết kế mức khái niệm

Mô hình thực thể kết hợp (ERD) của hệ thống được xây dựng xoay quanh các thực thể chính sau:



Hình 3.4. Sơ đồ quan hệ thực thể (ERD) của hệ thống EduTwin

3.5.2 Thiết kế chi tiết lược đồ dữ liệu

Hệ thống bao gồm 5 nhóm bảng chính: Quản trị người dùng, Cấu trúc động, Dữ liệu điểm số, Hỗ trợ AI và Nhật ký hội thoại.

- Nhóm bảng Cấu trúc động (Core Dynamic Engine)** Đây là trái tim của hệ thống, cho phép Admin định nghĩa bất kỳ mô hình giáo dục nào mà không cần thay đổi code.
- Nhóm bảng Dữ liệu điểm số** Dữ liệu điểm số được lưu trữ đọc (Normalized) để tối ưu cho các truy vấn thông kê, nhưng tham chiếu đến metadata động.
- Nhóm bảng Người dùng và Bảo mật (PII)** Lưu trữ thông tin định danh với cơ chế mã hóa cấp độ trường (Field-level Encryption).
- Nhóm bảng Hỗ trợ AI** Các bảng phục vụ cho việc theo dõi hoạt động AI, lưu trữ tài liệu tham khảo và cấu hình mô hình Machine Learning.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
structure_name	VARCHAR	Tên cấu trúc (VD: "THPT Quốc Gia 2024").
num_time_points	INTEGER	Số lượng mốc thời gian.
num_subjects	INTEGER	Số lượng môn học.
time_point_labels	JSONB	Danh sách mốc thời gian. VD: ["HK1-L10", "HK2-L10"].
subject_labels	JSONB	Danh sách môn học. VD: ["Toán", "Lý", "Anh"].
scale_type	VARCHAR	Thang điểm (VD: "0-10", "0-100", "A-F", "GPA").
pipeline_enabled	BOOLEAN	Bật/tắt pipeline dự đoán ML.
is_active	BOOLEAN	Cờ đánh dấu cấu trúc đang được sử dụng hiện hành.
created_at	TIMESTAMP	Thời điểm tạo.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.5. Bảng custom_teaching_structures - Định nghĩa cấu trúc học tập

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
user_id	INTEGER (FK)	Tham chiếu bảng Users.
structure_id	INTEGER (FK)	Tham chiếu bảng Structures.
subject	VARCHAR	Tên môn học (phải tồn tại trong subject_labels).
time_point	VARCHAR	Mốc thời gian (phải tồn tại trong time_point_labels).
actual_score	FLOAT	Điểm thực tế (có thể NULL nếu chưa có).
predicted_score	FLOAT	Điểm dự đoán bởi AI.
predicted_source	VARCHAR	Thuật toán dùng để dự đoán ('knn', 'kernel_regression', 'lwlr').
predicted_status	VARCHAR	Trạng thái dự đoán ('active', 'replaced').
created_at	TIMESTAMP	Thời điểm tạo.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.6. Bảng custom_user_scores - Lưu trữ điểm số người dùng

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
structure_id	INTEGER (FK)	Tham chiếu cấu trúc tương ứng.
sample_name	VARCHAR	Tên mẫu dữ liệu (tùy chọn).
score_data	JSONB	Vector điểm số đầy đủ của một mẫu. VD: {"Toán_HK1": 8.0, "Lý_HK1": 7.5}.
metadata	JSONB	Thông tin bổ sung về mẫu.
created_at	TIMESTAMP	Thời điểm tạo.

Bảng 3.7. Bảng custom_reference_dataset - Dữ liệu mẫu (Training Data)

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
username	VARCHAR (Unique)	Tên đăng nhập.
hashed_password	VARCHAR	Mật khẩu đã băm (Bcrypt).
first_name	VARCHAR	Tên người dùng.
last_name	VARCHAR	Họ người dùng.
email	VARCHAR	Mã hóa Envelope Encryption.
phone	VARCHAR	Mã hóa Envelope Encryption.
address	VARCHAR	Mã hóa Envelope Encryption.
age	VARCHAR	Tuổi người dùng.
current_grade	VARCHAR	Khối lớp hiện tại.
role	VARCHAR	Vai trò ('user', 'admin').
preferences	JSONB	Lưu sở thích học tập, phong cách chat.
uploaded_documents	JSON	Danh sách các tài liệu của người dùng.

Bảng 3.8. Bảng users - Thông tin người dùng

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
user_id	INTEGER (FK)	Tham chiếu bảng Users.
structure_id	INTEGER (FK)	Tham chiếu cấu trúc học tập.
insight_type	VARCHAR	Loại nhận định ('slide_comment', 'chat_response', 'subject_analysis').
context_key	VARCHAR	Khóa ngữ cảnh ('overview_chart', 'Math', 'A00').
content	TEXT	Nội dung nhận định của AI.
metadata	JSONB	Thông tin bổ sung (điểm liên quan, độ tin cậy).
created_at	TIMESTAMP	Thời điểm tạo.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.9. Bảng ai_insights - Lưu trữ nhận định AI về học sinh

5. Nhóm bảng Hội thoại (Chat History) Lưu trữ toàn bộ lịch sử tương tác giữa học sinh và AI.

Thiết kế này cho phép học sinh có nhiều phiên hội thoại độc lập, mỗi phiên theo dõi một chủ đề hoặc mục tiêu học tập khác nhau. Đồng thời hỗ trợ quản lý preferences riêng biệt cho từng cấu trúc học tập mà user tham gia.

3.5.3 Phân tích các quyết định thiết kế quan trọng

A. Tại sao sử dụng JSONB cho Cấu trúc động? Thay vì sử dụng mô hình Entity–Attribute–Value (EAV) truyền thống vốn phức tạp trong truy vấn và khó bảo trì, đề tài lựa chọn lưu trữ định nghĩa môn học và học kỳ dưới dạng JSONB.

- **Hiệu năng:** PostgreSQL hỗ trợ Generalized Inverted Index (GIN) cho JSONB, cho phép truy xuất dữ liệu metadata với độ phức tạp $O(1)$ thay vì phải JOIN hàng loạt bảng như EAV.
- **Linh hoạt:** Khi nhà trường thay đổi chương trình học (ví dụ: thêm môn "Trí tuệ nhân tạo"), Admin chỉ cần cập nhật mảng JSON trong bảng structures mà không cần thực hiện lệnh ALTER TABLE nặng nề hay downtime hệ thống.

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
structure_id	INTEGER (FK)	Tham chiếu cấu trúc học tập.
file_name	VARCHAR	Tên file gốc.
file_type	VARCHAR	Loại file ('pdf', 'docx', 'txt').
file_size	INTEGER	Kích thước file (bytes).
original_content	TEXT	Nội dung đầy đủ được trích xuất.
extracted_summary	TEXT	Nội dung tóm tắt để bổ sung context cho LLM.
extraction_method	VARCHAR	Phương pháp trích xuất ('llm_summary').
metadata	JSONB	Thông tin bổ sung (số trang, sections).
uploaded_by	INTEGER (FK)	Người tải lên (tham chiếu Users).
created_at	TIMESTAMP	Thời điểm tạo.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.10. Bảng custom_structure_documents - Lưu tài liệu tham khảo cho AI

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
knn_n	INTEGER	Số láng giềng K cho thuật toán KNN.
kr_bandwidth	FLOAT	Bảng thông cho Kernel Regression.
lwlr_tau	FLOAT	Tham số Tau cho LWLR.
updated_by	INTEGER (FK)	Người cập nhật (tham chiếu Users).
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.11. Bảng ml_model_parameters - Tham số các mô hình ML

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
active_model	VARCHAR	Mô hình đang được sử dụng ('knn', 'kernel_regression', 'lwlr').
updated_by	INTEGER (FK)	Người cập nhật (tham chiếu Users).
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.12. Bảng ml_model_config - Cấu hình mô hình ML đang hoạt động

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
user_id	INTEGER (FK)	Tham chiếu bảng Users.
title	VARCHAR	Tiêu đề phiên (tự động sinh từ tin nhắn đầu).
mode	STRING	Phân biệt phiên trò chuyện và phiên học tập
created_at	TIMESTAMP	Thời điểm tạo phiên.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.13. Bảng chat_sessions - Quản lý phiên hội thoại

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
session_id	INTEGER (FK)	Tham chiếu bảng ChatSessions.
role	VARCHAR	Vai trò: 'user', 'assistant', hoặc 'system'.
content	TEXT	Nội dung tin nhắn.
metadata	JSONB	Thông tin bổ sung về tin nhắn.
created_at	TIMESTAMP	Thời điểm gửi.

Bảng 3.14. Bảng chat_messages - Lưu trữ tin nhắn

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Tên trường	Kiểu dữ liệu	Mô tả
id	SERIAL (PK)	Khóa chính.
user_id	INTEGER (FK)	Tham chiếu bảng Users.
structure_id	INTEGER (FK)	Tham chiếu bảng Structures.
current_timepoint	VARCHAR	Mốc thời gian hiện tại của user cho structure này.
created_at	TIMESTAMP	Thời điểm tạo.
updated_at	TIMESTAMP	Thời điểm cập nhật gần nhất.

Bảng 3.15. Bảng user_structure_preferences - Lưu trữ preferences của user theo structure

B. Chiến lược Đánh chỉ mục (Indexing Strategy) Để đảm bảo tốc độ phản hồi < 200ms cho các truy vấn dự đoán, hệ thống thiết lập các chỉ mục tối ưu:

```
-- Index duy nhất để đảm bảo tính toàn vẹn dữ liệu điểm số
CREATE UNIQUE INDEX ix_user_score_unique
ON custom_user_scores(user_id, structure_id, subject, time_point);

-- GIN Index cho phép tìm kiếm nhanh trong dữ liệu huấn luyện AI
CREATE INDEX ix_ref_data_json
ON custom_reference_dataset USING GIN (score_data);
```

C. Cơ chế Bảo mật Dữ liệu (Data Security) Khác với các hệ thống thông thường chỉ mã hóa mật khẩu, EduTwin thực hiện mã hóa hai chiều (Symmetric Encryption) đối với các trường PII (Email, Phone) ngay tại tầng ứng dụng (Application Layer) trước khi lưu xuống DB. Điều này đảm bảo rằng ngay cả khi Database Administrator truy cập trực tiếp vào cơ sở dữ liệu cũng không thể đọc được thông tin cá nhân của học sinh.

3.6 Thiết kế Thuật toán và Luồng xử lý dữ liệu

Để đảm bảo khả năng dự đoán chính xác trong thời gian thực (real-time) với dữ liệu không đầy đủ, hệ thống EduTwin triển khai một pipeline xử lý dữ liệu phức hợp. Phần này mô tả chi tiết logic vận hành của các thuật toán cốt lõi.

3.6.1 Luồng xử lý dữ liệu khuyết

Trong môi trường thực tế, việc áp dụng trực tiếp công thức Euclidean chuẩn (như đã trình bày ở Chương 2) là không khả thi do sự hiện diện của các giá trị NaN. Hệ thống EduTwin giải quyết bằng thuật toán KNN Imputation được cải tiến (Modified KNN Imputer):

1. **Định nghĩa khoảng cách thích ứng (NaN-aware Distance):** Để so sánh vector x (có dữ liệu khuyết) và vector y , hệ thống sử dụng công thức khoảng cách hiệu chỉnh nhằm bù đắp cho các chiều dữ liệu bị thiếu:

$$d(x, y) = \begin{cases} \infty & \text{nếu } P = \emptyset \\ \sqrt{\frac{N}{|P|} \sum_{i \in P} (x_i - y_i)^2} & \text{nếu } P \neq \emptyset \end{cases}$$

Trong đó: N là tổng số chiều dữ liệu, P là tập hợp các chỉ số môn học mà cả x và y đều có điểm số. Hệ số $\frac{N}{|P|}$ đóng vai trò "phóng đại" khoảng cách tính được trên tập con P để ước lượng khoảng cách trên toàn không gian N chiều.

2. **Tìm kiếm láng giềng và Điền khuyết:** Hệ thống chọn ra k láng giềng tốt nhất (\mathcal{N}_k) dựa trên khoảng cách thích ứng trên và thực hiện điền khuyết trọng số:

$$\hat{x}_j = \frac{\sum_{y \in \mathcal{N}_k} w_y \cdot y_j}{\sum_{y \in \mathcal{N}_k} w_y} \quad \text{với} \quad w_y = \frac{1}{d(x, y) + 10^{-5}}$$

Trong đó hằng số nhỏ 10^{-5} để tránh lỗi chia cho 0.

3.6.2 Chiến lược Tối ưu hóa Dự đoán với Phân cụm K-Means và Tuyển chọn Mẫu hình

Để giải quyết bài toán độ phức tạp tính toán $O(N)$ của thuật toán Lazy Learning khi dữ liệu tham chiếu N tăng lớn (vẫn đề độ trễ), hệ thống áp dụng chiến lược hai tầng "Phân cụm Toàn cục - Hồi quy Cục bộ".

Tầng 1: Đánh chỉ mục bằng Phân cụm (Global Indexing)

- Sử dụng thuật toán **K-Means** (thư viện Scikit-learn) để phân hoạch không gian dữ liệu thành K cụm (Clusters). Số lượng cụm K được xác định động dựa trên quy mô

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

tập dữ liệu N theo công thức:

$$K = \begin{cases} 1 & \text{nếu } N < 3000 \\ \min\left(\left\lceil \frac{N}{3000} \right\rceil, 100\right) & \text{nếu } N \geq 3000 \end{cases} \quad (3.1)$$

Thiết kế này đảm bảo mỗi cụm có khoảng 3,000 mẫu, đủ cho các thuật toán Lazy Learning hoạt động ổn định. Khi $N < 3000$, hệ thống gán $K = 1$ để ưu tiên độ chính xác tuyệt đối trên tập dữ liệu nhỏ.

- Tại mỗi cụm, hệ thống **lưu trữ toàn bộ điểm dữ liệu** (được sắp xếp theo khoảng cách đến tâm cụm) để cho phép tuyển chọn mẫu hình thích ứng tại thời điểm dự đoán.
- Kết quả: Một bộ chỉ mục (Index) được lưu trữ trên RAM (qua Redis) để truy xuất cực nhanh.

Tầng 2: Dự đoán Cục bộ với Tuyển chọn Mẫu hình Thích ứng (Adaptive Prototype Selection)

Tại thời điểm runtime (khi người dùng yêu cầu dự đoán), quy trình diễn ra như sau:

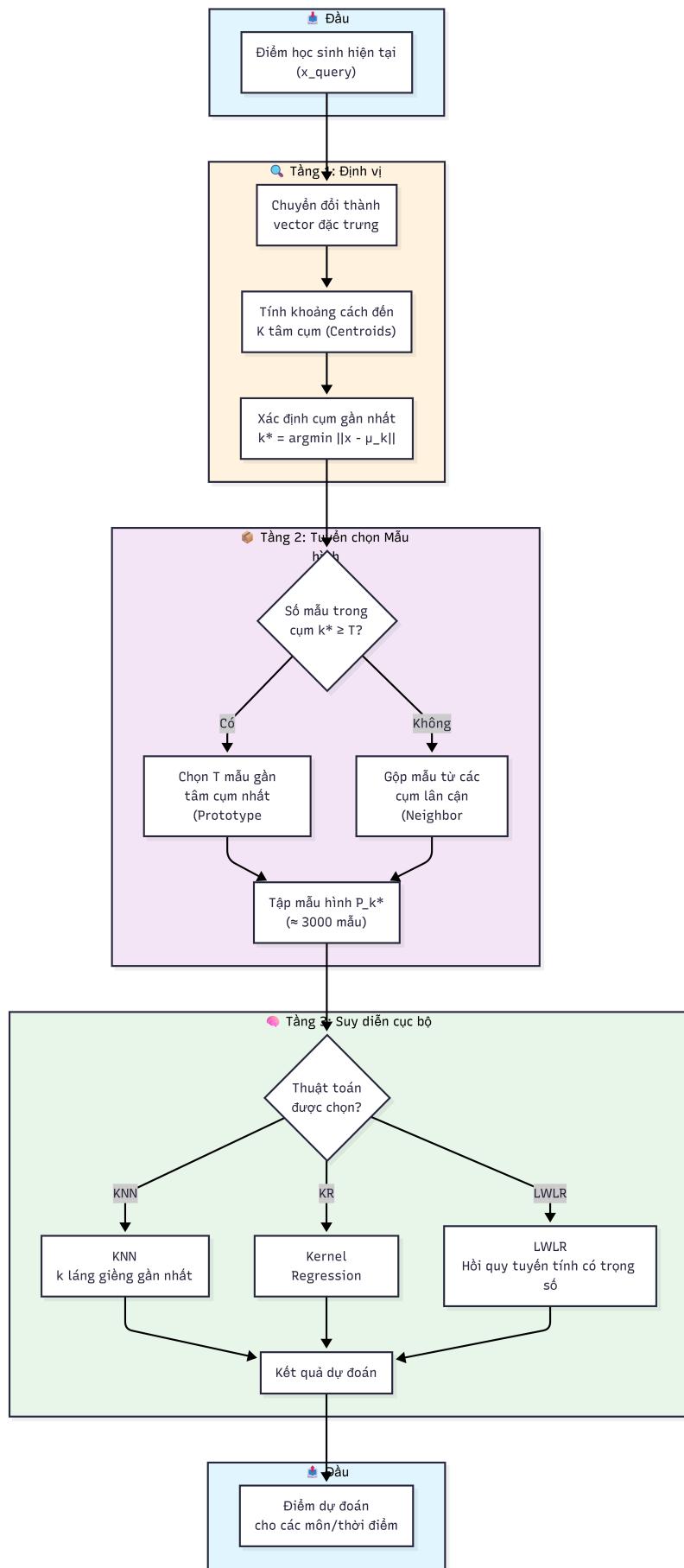
- *Bước 1 (Assignment):* Xác định cụm k^* mà vector học sinh x thuộc về bằng cách so sánh khoảng cách tới các tâm cụm (Centroids). Độ phức tạp giảm từ $O(N)$ xuống $O(K)$.
- *Bước 2 (Adaptive Retrieval):* Truy xuất mẫu từ cụm k^* với chiến lược thích ứng:
 - ◊ Nếu $|C_{k^*}| \geq T$: Chọn $T = 3000$ mẫu gần tâm cụm nhất (Prototype Selection).
 - ◊ Nếu $|C_{k^*}| < T$: Gộp thêm mẫu từ các cụm lân cận theo thứ tự khoảng cách centroid (Neighbor Merging).
- *Bước 3 (Lazy Learning Inference):* Chạy thuật toán hồi quy chính (KNN, KR, LWLR) trên tập mẫu hình P_{k^*} (luôn có khoảng $T = 3000$ mẫu) để đưa ra kết quả cuối cùng.

3.6.3 Luồng dự đoán Lazy Learning

Quy trình dự đoán được thiết kế để tối ưu hóa thời gian phản hồi bằng cách chia nhỏ không gian tìm kiếm. Thay vì quét toàn bộ cơ sở dữ liệu (Global Scan), hệ thống thực hiện quy trình 3 bước sau: Chi tiết thuật toán vận hành như sau:

Bước 1: Định vị (Localization)

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG



Hình 3.5. Sơ đồ luồng xử lý dự đoán Lazy Learning với Phân cụm K-Means và Tuyển chọn Mẫu hình Thích ứng

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- Hệ thống chuyển đổi bảng điểm hiện tại của học sinh thành vector đặc trưng x_{query} .
- Tính khoảng cách Euclidean từ x_{query} tới các tâm cụm (Centroids) đã được huấn luyện trước bằng K-Means.
- Xác định cụm k^* có khoảng cách nhỏ nhất: $k^* = \arg \min_k \|x_{query} - \mu_k\|^2$.

Bước 2: Tuyển chọn Mẫu hình Thích ứng (Adaptive Prototype Selection)

- Tải tập mẫu của cụm k^* từ Redis Cache (các mẫu đã được sắp xếp theo khoảng cách đến tâm).
- Nếu cụm có $\geq T$ mẫu ($T = 3000$): Chọn T mẫu đầu tiên (gần tâm nhất).
- *Cơ chế Neighbor Merging*: Nếu cụm có $< T$ mẫu, hệ thống tự động mở rộng phạm vi tìm kiếm sang các cụm lân cận (theo thứ tự khoảng cách centroid) cho đến khi thu thập đủ T mẫu để đảm bảo độ ổn định thống kê.

Bước 3: Suy diễn cục bộ (Local Inference) Tùy thuộc vào cấu hình của Admin, hệ thống áp dụng một trong ba thuật toán trên tập mẫu hình cục bộ (luôn có khoảng $T = 3000$ mẫu):

- **KNN**: Lấy trung bình của k láng giềng gần nhất (thường chọn $k = 5 \sim 15$) trong tập mẫu hình.
- **KR**: Tính trung bình có trọng số sử dụng hàm Gaussian Kernel với tham số độ rộng dài h .
- **LWLR**: Tối ưu hóa hàm mất mát cục bộ để tìm tham số θ cho phương trình tuyến tính, sử dụng tham số độ rộng dài τ .

3.6.4 Luồng xử lý AI Chatbot (Chat Mode)

Quy trình xử lý của Chatbot được thiết kế để đảm bảo tính cá nhân hóa và bảo mật dữ liệu:

1. Context Building: Hệ thống thu thập ngữ cảnh từ 3 nguồn:

- Dữ liệu điểm số và dự đoán từ Digital Twin.
- Sở thích đã học (learned preferences).
- 20 tin nhắn gần nhất (Sliding Window).

2. PII Redaction: Lớp bảo mật ẩn danh hóa thông tin cá nhân trước khi gửi đến LLM:

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- Họ tên → SHA-256 hash (VD: USER_A1B2C3).
- Email → Mask (VD: n***n@g***l.com).
- SĐT → Mask (VD: PHONE_XXXX5678).

3. **LLM Request:** Gửi trực tiếp đến Gemini API với System Prompt định hướng vai trò "Cố vấn Học tập" (Academic Mentor).

4. **Hybrid Personalization:** Phân tích response để học sở thích người dùng:

- Keyword Detection: Quét 250+ từ khóa tiếng Việt.
- LLM Analysis: Trigger khi đủ 8 meaningful messages.
- Cập nhật preferences vào user profile.

3.6.5 Luồng xử lý Learning Agent (Learning Mode)

Learning Agent sử dụng kiến trúc ReAct (Reasoning + Acting) để giải đáp câu hỏi học tập phức tạp:

1. **Auto Document Search:** Agent tự động tìm kiếm trong tài liệu người dùng đã upload trước khi bắt đầu suy luận.

2. **Vòng lặp ReAct** (tối đa 5 iterations):

- **Thought:** LLM phân tích vấn đề và lập kế hoạch.
- **Action:** Chọn và thực thi tool phù hợp:
 - Calculator: Tính toán biểu thức (numexpr).
 - Wikipedia: Tra cứu kiến thức (tiếng Việt).
 - Python REPL: Thực thi code trong sandbox.
 - Document Search: Tìm kiếm tài liệu user.
- **Observation:** Nhận kết quả từ tool.

3. **Self-Reflection:** Agent tự đánh giá câu trả lời (đủ thông tin? chính xác?) trước khi phản hồi.

4. **Real-time Updates:** Gửi trạng thái suy luận về frontend qua WebSocket (Socket.IO) để hiển thị tiến trình.

3.7 Thiết kế Giao diện người dùng

Giao diện người dùng được thiết kế theo phong cách hiện đại (Material/Flat Design), tối ưu hóa cho trải nghiệm trên cả máy tính và thiết bị di động, tập trung vào khả năng trực quan hóa dữ liệu.

3.7.1 Sơ đồ tổ chức thông tin

Hệ thống được tổ chức thành các luồng chức năng rõ ràng:

3.7.2 Thiết kế chi tiết các màn hình chính

1. Màn hình Dashboard (DataViz) Đây là màn hình trung tâm, cung cấp cái nhìn toàn cảnh về năng lực học sinh.

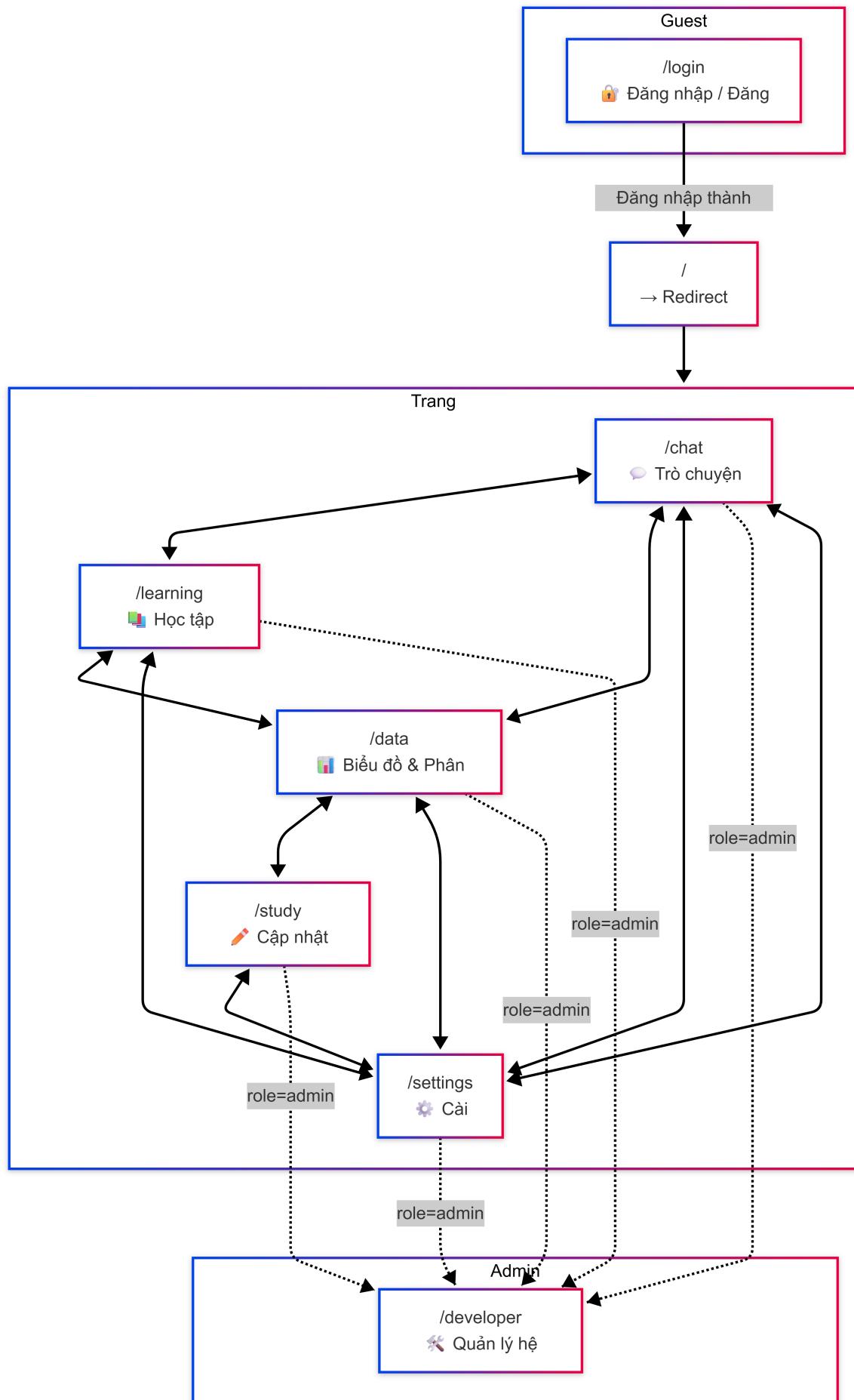
- **Radar Chart:** Biểu đồ mạng nhện đa giác, mỗi đỉnh đại diện cho một môn học. Vùng diện tích phủ giúp học sinh nhận diện nhanh thiên hướng (lệch về Tự nhiên hay Xã hội).
- **Bar Chart:** Biểu đồ cột hiển thị điểm số theo từng môn học.
- **Line Chart (Dự đoán):** Biểu đồ đường thể hiện hai dòng dữ liệu: đường nét liền (điểm thực tế quá khứ) và đường nét đứt (điểm dự đoán tương lai).
- **AI Insight Card:** Một thẻ thông báo nổi bật chứa nhận định ngắn gọn từ AI (ví dụ: "Bạn đang có xu hướng giảm điểm nhẹ ở môn Lý, hãy chú ý!").

2. Màn hình Chatbot AI Giao diện hội thoại được thiết kế tương tự các ứng dụng nhắn tin hiện đại:

- **Khu vực hội thoại:** Hiển thị tin nhắn dạng bong bóng (bubbles). Tin nhắn của AI hỗ trợ hiển thị Markdown (đậm, nghiêng, danh sách) để trình bày lộ trình học tập rõ ràng.
- **Chat Session Sidebar:** Hiển thị các chat session

3. Màn hình Learning Agent (Learning Mode) Giao diện học tập nâng cao với AI Agent có khả năng suy luận:

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG



Hình 3.6. Sơ đồ tổ chức chức năng (Site Map) của ứng dụng EduTwin

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

- **Khu vực hội thoại:** Hiển thị câu hỏi và phản hồi, hỗ trợ Markdown và code highlighting cho nội dung kỹ thuật.
- **Reasoning Panel:** Hiển thị real-time các bước suy luận của Agent:
 - ◊ Thought: Quá trình phân tích vấn đề
 - ◊ Action: Tool đang được sử dụng
 - ◊ Observation: Kết quả từ tool
- **Document Manager:** Cho phép upload tài liệu (PDF, DOCX, TXT) để Agent tìm kiếm và trích dẫn.
- **Tool Indicators:** Hiển thị icons và trạng thái của 4 tools (Calculator, Wikipedia, Python, Document Search).
- **Self-Reflection Badge:** Hiển thị kết quả tự đánh giá của Agent (đủ thông tin, cần tìm thêm, v.v.).

4. Màn hình Công cụ Quản trị (Developer Tools) Dành riêng cho Admin để thao tác với các tính năng Cấu trúc động:

- **Structure Editor:** Giao diện cho phép thêm/bớt môn học và học kỳ dưới dạng danh sách động, sau đó xuất ra JSON lưu vào Database.
- **Dataset Manager:** Khu vực kéo thả file Excel để tải lên dữ liệu huấn luyện mới.
- **Model Tuning:** Các thanh trượt (Sliders) để điều chỉnh tham số k (số láng giềng) hoặc Bandwidth h (độ rộng dải) và xem kết quả đánh giá (RMSE) thay đổi tức thì.

3.8 Tổng hợp Công nghệ và Môi trường Triển khai

EduTwin được xây dựng dựa trên kiến trúc Modular Monolith hiện đại, phân tách rõ ràng giữa tầng xử lý dữ liệu thông minh (Backend) và tầng giao diện tương tác người dùng (Frontend). Các công nghệ được lựa chọn đều nhằm phục vụ mục tiêu tối thượng: đảm bảo tốc độ phản hồi thời gian thực cho thuật toán Lazy Learning.

3.8.1 Danh sách Công nghệ

- **Backend (Python Ecosystem):**

- **FastAPI:** Framework chủ đạo xây dựng REST API. Được chọn nhờ khả năng xử lý bất đồng bộ (Async I/O), cho phép hệ thống phục vụ 40-50 người dùng đồng thời với cơ chế Semaphore kiểm soát concurrency.
- **Scikit-learn & NumPy:** "Bộ nào" tính toán của hệ thống, thực thi K-Means Clustering, KNN Imputation và các thuật toán Lazy Learning.
- **LangChain:** Framework xây dựng AI Agent, tích hợp Google Gemini API cho cả Chatbot và Learning Agent.
- **Socket.IO:** WebSocket cho real-time communication, hiển thị tiến trình suy luận của Agent.
- **SQLAlchemy 2.0:** ORM hiện đại ánh xạ linh hoạt đối tượng Python với dữ liệu JSONB.

- **Frontend (React Ecosystem):**

- **React 19 & Vite 7:** Phiên bản React mới nhất với Concurrent Rendering. Vite tối ưu tốc độ build và Hot Module Replacement.
- **Recharts & Framer Motion:** Thư viện trực quan hóa biểu đồ Radar, Bar, Line với hiệu ứng animation sinh động.
- **Socket.IO Client:** Nhận real-time updates từ backend.

- **Database & Caching:**

- **PostgreSQL 15.7:** Cơ sở dữ liệu chính với JSONB cho cấu trúc học tập động.
- **Redis 7.0:** In-memory cache cho Cluster Index, Prediction Cache, Session management. TTL: Prediction (1h), Evaluation (2h), Cluster (24h).

- **AI & LLM:**

- **Google Gemini 2.5 Flash:** LLM cho Chatbot và Learning Agent.
- **Concurrency Control:** Semaphore(10) giới hạn 10 request đồng thời.
- **PII Redaction:** Fernet (AES-128) encryption + SHA-256 hashing.

3.8.2 Môi trường Triển khai

1. Môi trường Phát triển (Development)

- **OS:** Windows 11 hoặc macOS.
- **Công cụ:** VS Code, Docker Desktop, Postman.
- **Local Stack:** Docker Compose với PostgreSQL, Redis, Backend, Frontend.
- **Quản lý mã nguồn:** Git & GitHub với CI/CD (GitHub Actions).

2. Môi trường Vận hành (Production - AWS)

- **Compute - ECS Fargate:**

- ◊ Backend Task: 1 vCPU, 2GB RAM, 4 Uvicorn workers.
 - ◊ Frontend Task: 0.25 vCPU, 0.5GB RAM (Nginx).

- **Database - RDS PostgreSQL 15.7:** db.t3.small, 20GB SSD.

- **Cache - ElastiCache Redis 7.0:** cache.t3.small.

- **Networking:**

- ◊ Application Load Balancer (ALB) với HTTPS/SSL.
 - ◊ Route 53 DNS.
 - ◊ ACM Certificate Manager.

- **Security:** AWS Secrets Manager cho API keys và credentials.

- **LLM Integration:** Google Gemini API qua HTTPS, với lớp PII Redaction trung gian tại ECS Backend.

Kiến trúc này đảm bảo sự cân bằng giữa chi phí vận hành (sử dụng Fargate Spot, không cần GPU) và hiệu năng (tận dụng LLM của Google qua API).

Chương 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

4.1 Môi trường và Dữ liệu thực nghiệm

4.1.1 Môi trường thực nghiệm

Để đảm bảo tính khách quan và hiệu quả, nghiên cứu được thực hiện trên hai môi trường tính toán riêng biệt phục vụ cho hai giai đoạn: huấn luyện mô hình và triển khai ứng dụng.

Đối với quá trình Huấn luyện và Đánh giá (Training & Evaluation): Việc so sánh hiệu năng các mô hình (đặc biệt là Kịch bản 2) đòi hỏi tài nguyên tính toán lớn để xử lý các thuật toán phức tạp như Long Short-Term Memory (LSTM) hay Ensemble Learning. Do đó, chúng tôi sử dụng nền tảng đám mây Kaggle Notebooks với cấu hình hiệu năng cao. Hệ thống được trang bị kép Graphics Processing Unit (GPU) NVIDIA Tesla T4 (Tổng VRAM 32GB) nhằm tăng tốc huấn luyện cho các mô hình Eager Learning, kết hợp với 30GB RAM để đảm bảo khả năng xử lý in-memory toàn bộ ma trận dữ liệu. Vi xử lý trung tâm là Intel Xeon (2 vCPUs @ 2.20GHz) với thời gian phiên làm việc liên tục lên đến 12 giờ.

Đối với các quá trình khác: Mục tiêu là mô phỏng điều kiện thực tế tại các trường học có hạ tầng hạn chế. Thiết bị thử nghiệm là Laptop Lenovo ThinkPad T490 với cấu hình khiêm tốn: vi xử lý Intel Core i5-10210U (dòng tiết kiệm điện ULV), 8GB RAM và không sử dụng GPU rời. Nền tảng LLM được tích hợp thông qua API của Google Gemini-2.5-Flash nhằm giảm tải tính toán cho thiết bị đầu cuối.

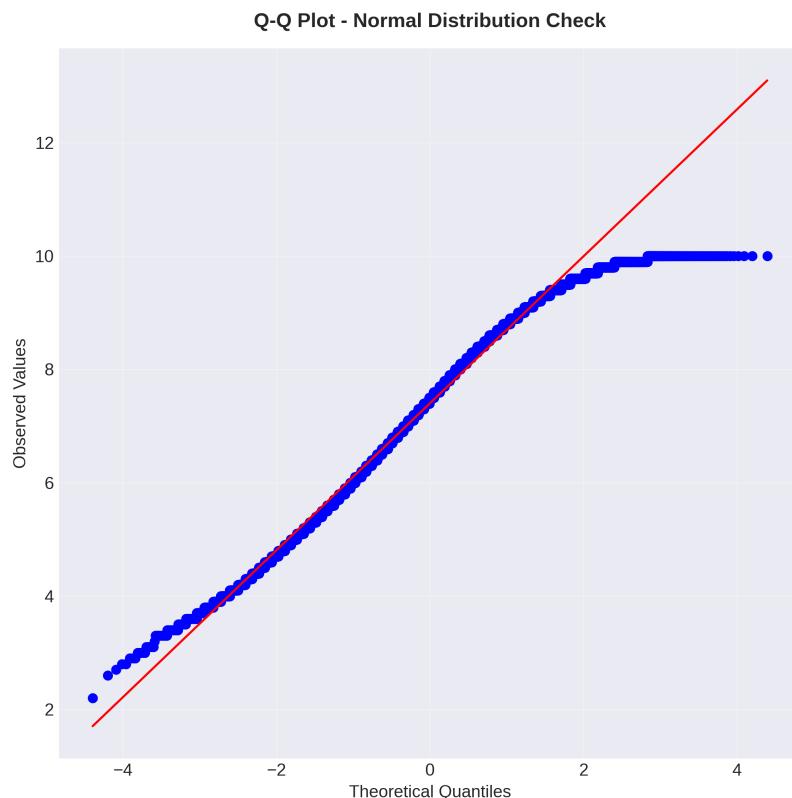
4.1.1.1 Phân tích Khám phá Dữ liệu

Tập dữ liệu thực nghiệm của hệ thống EduTwin được tổng hợp từ hồ sơ học tập chính thức tại Trường THPT Nguyễn Khuyến trong giai đoạn kéo dài bảy năm, từ 2017 đến 2024. Dữ liệu bao gồm điểm số của chín môn học cốt lõi trong chương trình trung học phổ thông (Toán, Ngữ văn, Tiếng Anh, Vật lý, Hóa học, Sinh học, Lịch sử, Địa lý và Giáo dục công dân), được ghi nhận xuyên suốt ba năm học (lớp 10, 11 và 12), với hai học kỳ mỗi năm. Sau quá trình tiền xử lý, loại bỏ các giá trị khuyết và các bản ghi không hợp lệ, tập dữ liệu cuối cùng bao gồm tổng cộng 123 291 điểm số thành phần, tương ứng với 2 283 hồ sơ học sinh hợp lệ.

Đặc điểm phân phối dữ liệu. Phân tích thống kê mô tả cho thấy phân phối điểm số tổng thể có xu hướng tiệm cận phân phối chuẩn, với giá trị trung bình đạt 7.40 và trung vị là 7.50. Khoảng chênh lệch nhỏ giữa hai thước đo xu hướng trung tâm này phản ánh

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

mức độ đối xứng tương đối cao của phân phối quanh giá trị trung bình. Kết quả kiểm định Shapiro–Wilk được thực hiện trên các tập con đại diện theo từng môn học cho thấy giả thuyết phân phối chuẩn bị bác bỏ về mặt thống kê ($p\text{-value} < 0.05$), tuy nhiên hiện tượng này là phổ biến đối với các tập dữ liệu có kích thước lớn. Do đó, việc đánh giá trực quan thông qua biểu đồ Q–Q đóng vai trò quan trọng hơn trong việc xem xét tính gần chuẩn của dữ liệu.



Hình 4.1. Biểu đồ Q–Q kiểm định phân phối chuẩn, minh họa mức độ tuân thủ của dữ liệu điểm số đối với phân phối chuẩn lý thuyết.

Biểu đồ Q–Q được trình bày tại Hình 4.1 cho thấy các điểm quan sát bám sát đường chuẩn lý thuyết tại các phân vị trung tâm, trong khi chỉ xuất hiện các sai lệch nhẹ ở hai đầu phân phối. Hiện tượng này chủ yếu bắt nguồn từ tính chất bị chặn của thang điểm (0–10), qua đó xác nhận rằng giả định phân phối gần chuẩn là hợp lý và có thể chấp nhận được cho các phân tích mô hình hóa tiếp theo.

Biến thiên hiệu năng theo môn học. Sự khác biệt về mức độ đạt được giữa các môn học được thể hiện rõ ràng trong Hình 4.2(b). Nhóm các môn thuộc khối Khoa học Xã hội và kỹ năng công dân ghi nhận mức điểm trung bình cao hơn đáng kể, trong đó Giáo dục công dân đạt giá trị cao nhất (8.48), tiếp theo là Địa lý (7.79) và Lịch sử (7.44). Ngược lại, các môn học mang tính nền tảng và tư duy trừu tượng cao hơn như Toán học (6.93), Ngữ

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

văn (6.99) và Vật lý (7.14) có điểm trung bình thấp hơn một cách nhất quán. Khoảng cách lên tới 1.55 điểm giữa môn có điểm trung bình cao nhất và thấp nhất phản ánh sự khác biệt về độ khó nội dung, phương pháp đánh giá cũng như đặc thù giảng dạy giữa các tổ bộ môn trong cùng một môi trường giáo dục.

Phân phối xếp loại học lực. Phân tích phân phối điểm số theo các mức xếp loại học lực, được minh họa trong Hình 4.2(c), cho thấy một xu hướng lệch phải tích cực, với mật độ điểm tập trung chủ yếu ở các mức thành tích trung bình khá đến cao. Cụ thể, 75.4% tổng số điểm thuộc hai nhóm “Khá” (6.5–8.0, chiếm 38.3%) và “Giỏi” (8.0–10.0, chiếm 37.1%). Ngược lại, tỷ lệ điểm dưới trung bình (nhỏ hơn 5.0) chỉ chiếm 3.6%, trong đó mức “Kém” gần như không đáng kể (0.1%). Sự tập trung cao của điểm số ở dải trên, cùng với số lượng rất hạn chế các quan sát ngoại lai ở mức thấp, cho thấy chất lượng đầu vào tương đối đồng đều cũng như hiệu quả đào tạo ổn định của Trường THPT Nguyễn Khuyến trong giai đoạn nghiên cứu.

4.2 Kịch bản 1: Đánh giá hiệu quả xử lý dữ liệu khuyết

Trước khi đưa vào mô hình dự đoán, tính toàn vẹn của dữ liệu là yếu tố tiên quyết. Kịch bản này tập trung so sánh hiệu quả của thuật toán đề xuất (KNN Imputation) so với các phương pháp điền khuyết thông kê cơ bản nhằm tìm ra giải pháp tối ưu cho bài toán dữ liệu giáo dục.

4.2.1 Thiết lập thực nghiệm

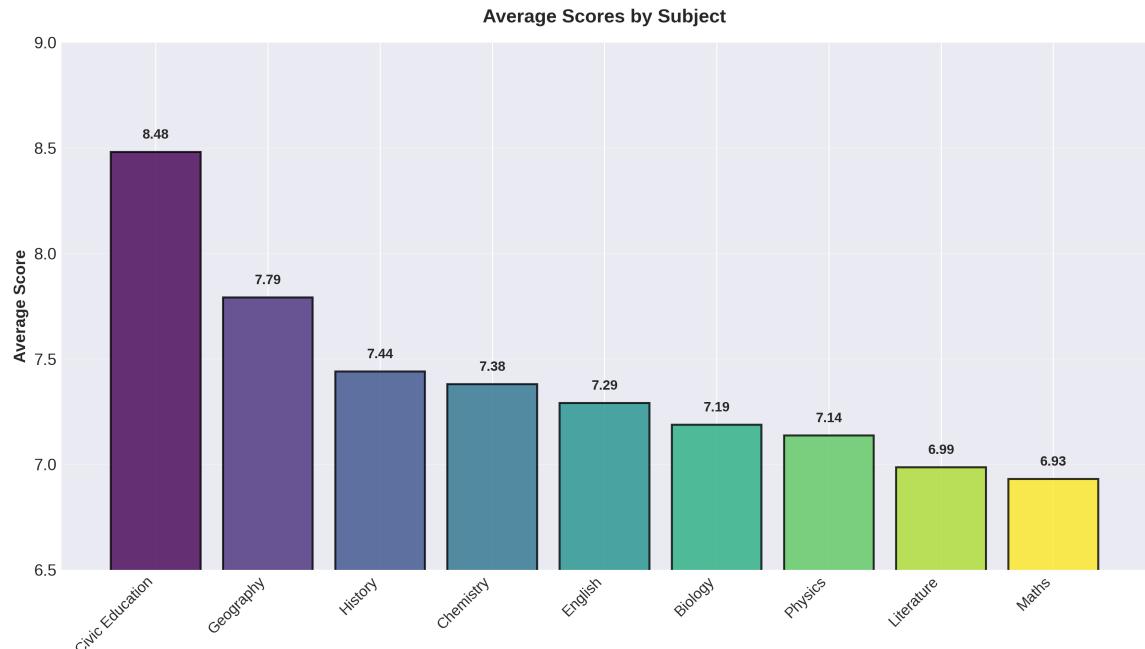
Dữ liệu gốc (đầy đủ) được áp dụng cơ chế xóa ngẫu nhiên Missing Completely At Random (MCAR) để tạo ra các tập dữ liệu khuyết giả lập với tỷ lệ từ 10% đến 50%. Nghiên cứu tiến hành so sánh ba phương pháp điền khuyết chính: Median Imputation (điền bằng trung vị), Mean Imputation (điền bằng trung bình), và phương pháp đề xuất KNN Imputation (điền dựa trên trung bình có trọng số của k láng giềng gần nhất).

4.2.2 Phân tích kết quả

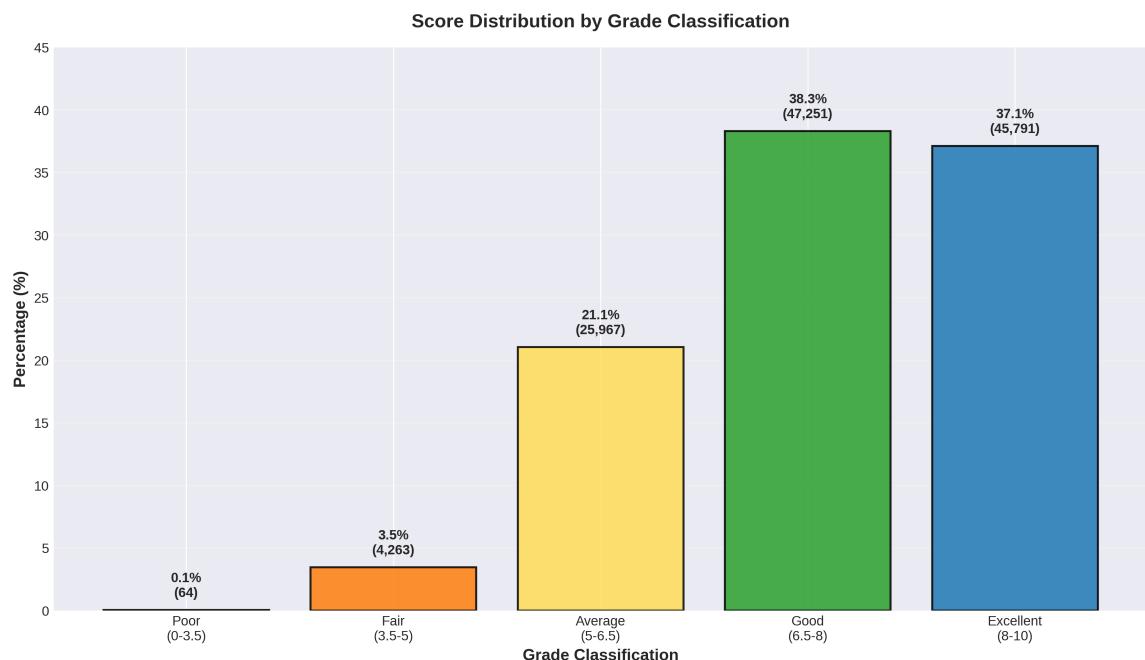
Kết quả thực nghiệm được trình bày tại Bảng 4.1 cho thấy sự vượt trội hoàn toàn của KNN Imputation trên mọi chỉ số đo lường.

Hạn chế của phương pháp thống kê: Các phương pháp Mean và Median Imputation cho kết quả R^2 rất thấp, chỉ đạt khoảng 0.17. Nguyên nhân cốt lõi là do các phương pháp này xử lý từng môn học một cách độc lập (univariate), bỏ qua hoàn toàn mối tương quan

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



(a) Điểm trung bình theo từng môn học



(b) Phân phối điểm số theo xếp loại

Hình 4.2. Phân tích mô tả dữ liệu học tập: (b) So sánh điểm trung bình giữa các môn học cho thấy sự khác biệt đáng kể về hiệu năng; và (c) Phân phối điểm số theo các mức xếp loại phản ánh sự tập trung rõ rệt ở nhóm thành tích cao.

	Tỷ lệ thiếu 10%			Tỷ lệ thiếu 50%		
	MAE	RMSE	R ²	MAE	RMSE	R ²
Median Imputation	0.97	1.20	0.17	0.95	1.18	0.17
Mean Imputation	0.97	1.19	0.17	0.96	1.18	0.17
KNN Imputation	0.52	0.67	0.73	0.60	0.78	0.64

Bảng 4.1. So sánh hiệu năng điền khuyết (KNN so với các phương pháp thống kê)

đa chiều giữa các môn học. Việc điền giá trị trung bình đơn thuần làm giảm phương sai (variance), khiến hồ sơ học sinh trở nên “trung bình hóa” và mất đi các đặc trưng phân loại cá nhân.

Sức mạnh của cấu trúc tương quan: Ngược lại, KNN Imputation đạt $R^2 \approx 0.73$ nhờ khả năng tái tạo điểm số dựa trên “hồ sơ láng giềng”. Cơ chế này hoạt động dựa trên nguyên lý tương đồng: khi điền điểm môn Lý bị thiếu, thuật toán sẽ tham chiếu điểm Toán, Hóa, Sinh của những học sinh có năng lực tương đương trong không gian dữ liệu, từ đó đưa ra ước lượng chính xác hơn.

Độ bền vững (Robustness): Đáng chú ý, ngay cả trong kịch bản cực đoan khi dữ liệu bị mất tới 50%, KNN vẫn duy trì được độ chính xác ấn tượng với $R^2 > 0.64$. Điều này chứng minh dữ liệu giáo dục có tính dư thừa thông tin (redundancy) cao – năng lực học sinh ở các môn học có sự liên kết chặt chẽ – và thuật toán đề xuất đã khai thác hiệu quả đặc tính này để phục hồi thông tin.

4.3 Kịch bản 2: Đánh giá hiệu năng đa mô hình — So sánh Lazy Learning và Eager Learning

Kịch bản này được thiết kế để kiểm chứng giả thuyết về tính hiệu quả của Lazy Learning trong bối cảnh giáo dục. Chúng tôi thực hiện đánh giá song song giữa ba thuật toán thuộc nhóm *Lazy Learning* (KNN, Kernel Regression, LWLR) và hai thuật toán *Eager Learning* hiện đại (XGBoost, LSTM). Bài toán đặt ra là dự đoán điểm học kỳ lớp 12 dựa trên dữ liệu lịch sử của 4 học kỳ lớp 10 và 11 trên tập dữ liệu gồm 2284 mẫu.

4.3.1 Kết quả thực nghiệm định lượng

Kết quả đánh giá trên tập kiểm thử (test set - chiếm 20% dữ liệu gốc, tương đương 457 mẫu) được tóm tắt trong Bảng 4.2. Các chỉ số đánh giá bao gồm Sai số tuyệt đối trung bình (MAE), Sai số bình phương trung bình (RMSE), Hệ số xác định (R^2) và Thời gian suy diễn.

Nhóm	Mô hình	MAE	RMSE	R^2
	KNN	0.53	0.68	0.66
<i>Lazy Learning</i>	LWLR	0.53	0.68	0.66
	KR	0.55	0.72	0.61
<i>Eager Learning</i>	LSTM	0.54	0.69	0.65
	XGBoost	0.54	0.69	0.64

Bảng 4.2. So sánh hiệu năng giữa các mô hình Lazy Learning và Eager Learning

4.3.2 Phân tích kết quả

Dữ liệu cho thấy một quan sát quan trọng: các mô hình Lazy Learning, dù đơn giản về cấu trúc, lại đạt hiệu năng **tương đương hoặc tốt hơn** so với các mô hình Eager Learning phức tạp. Nguyên nhân đến từ việc dữ liệu điểm số học sinh mang tính cấu trúc cục bộ rõ rệt. Những học sinh có hồ sơ điểm lớp 10–11 tương tự nhau thường có kết quả lớp 12 gần giống nhau. Các mô hình như KNN và LWLR tận dụng trực tiếp đặc tính này thông qua cơ chế tìm kiếm cục bộ, thay vì cố gắng học một hàm tổng quát toàn cục như XGBoost. Bên cạnh đó, LSTM tuy mạnh về chuỗi thời gian nhưng không phát huy được ưu thế do chuỗi dữ liệu đầu vào quá ngắn (chỉ 4 thời điểm), dẫn đến hiệu năng thấp hơn KNN dù chi phí tính toán lớn hơn rất nhiều. XGBoost cũng bộc lộ hạn chế khi xử lý các quan hệ dạng khoảng cách mượt mà của điểm số, dẫn đến chỉ số R^2 thấp hơn.

Tổng hợp lại, Lazy Learning không chỉ cạnh tranh sòng phẳng mà còn vượt trội Eager Learning trong bối cảnh cụ thể này. KNN đạt hiệu năng tốt nhất (MAE thấp nhất, R^2 cao nhất) trong cả 5 mô hình. Ngoài ra, KNN còn cung cấp khả năng giải thích cao (explainability) bằng việc truy xuất trực tiếp các học sinh “tương tự”, giúp tăng tính minh bạch cho hệ thống tư vấn.

4.4 Kịch bản 3: Đánh giá hiệu năng và Tốc độ phản hồi

Thách thức lớn nhất về mặt lý thuyết của Lazy Learning là tốc độ suy diễn chậm khi dữ liệu lớn (độ phức tạp $O(N)$). Kịch bản này nhằm kiểm chứng hiệu quả thực tế của giải pháp tối ưu hóa “Phân cụm K-Means và Tối ưu hóa Mẫu hình” đã được thiết kế trong chương 3.

Để kiểm chứng hiệu quả thực tế của chiến lược tối ưu hóa so với phương pháp quét toàn cục (*Global Scan*), nghiên cứu tiến hành thực nghiệm Stress Test trên tập dữ liệu mô phỏng quy mô lớn. Mục tiêu là đánh giá khả năng duy trì độ trễ thấp của hệ thống khi số lượng bản ghi tăng từ mức cơ bản lên đến 100.000 mẫu.

4.4.1 Thiết lập thực nghiệm và Dữ liệu

Thực nghiệm sử dụng tập dữ liệu gốc gồm 2.283 hồ sơ học sinh với 54 đặc trưng, bao gồm điểm số của 9 môn học tại 6 thời điểm đánh giá (từ học kỳ 1 lớp 10 đến học kỳ 2 lớp 12). Bài toán đặt ra là sử dụng 36 đặc trưng đầu vào (lớp 10 và 11) để dự đoán 18 đặc trưng mục tiêu (lớp 12) thông qua thuật toán KNN với tham số $k = 5$.

Để mô phỏng các môi trường dữ liệu lớn, dữ liệu gốc được nhân bản và làm giàu thông qua kỹ thuật thêm nhiễu Gaussian $\mathcal{N}(0, 0.3)$. Phương pháp này giúp tạo ra sự đa dạng cho dữ liệu sinh ra trong khi vẫn bảo toàn các đặc tính phân phối thống kê cốt lõi.

Cấu hình phân cụm cho phương pháp *Cached Query* được thiết lập động dựa trên quy mô dữ liệu N . Số lượng cụm K tuân theo công thức:

$$K = \min \left(\left\lceil \frac{N}{3000} \right\rceil, 100 \right),$$

nhằm đảm bảo kích thước mỗi cụm duy trì ổn định quanh ngưỡng 3.000 mẫu.

4.4.2 Phân tích kết quả thực nghiệm.

Kết quả định lượng được trình bày tại Bảng 4.3 cho thấy sự phân hóa rõ rệt về hiệu năng giữa hai phương pháp khi quy mô dữ liệu thay đổi.

Đối với *Global Scan*, thời gian xử lý tăng tuyến tính theo độ phức tạp $O(N)$. Tại ngưỡng $N = 100.000$, độ trễ hệ thống lên tới 839,02 ms, bắt đầu vượt quá ngưỡng chấp nhận cho các ứng dụng thời gian thực.

Ngược lại, phương pháp *Cached Query* thể hiện ưu thế vượt trội ở các tập dữ liệu lớn ($N \geq 10.000$). Nhờ cơ chế sàng lọc mẫu hình thích ứng (*Adaptive Prototype Selection*), thuật toán chỉ cần truy xuất khoảng $T \approx 3.000$ mẫu liên quan nhất cho mỗi truy vấn, thay

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

vì quét toàn bộ cơ sở dữ liệu. Kết quả là tại quy mô 100.000 bản ghi, hệ thống đạt tốc độ xử lý 72,99 ms, tương ứng với mức tăng tốc 11,50 lần so với phương pháp truyền thống.

Bảng 4.3. Kết quả Stress Test so sánh hiệu năng (dữ liệu thực tế, 54 features)

Quy mô N	Số cụm K	Global Scan (ms)	Cached Query (ms)	Tăng tốc
1.000	1	8,36	7,98	1,05×
10.000	4	68,95	37,97	1,82×
50.000	17	392,47	50,03	7,84×
100.000	34	839,02	72,99	11,50×

Đáng chú ý, ở quy mô nhỏ ($N < 3.000$), lợi ích của phân cụm là không đáng kể (tăng tốc ≈ 1) do toàn bộ dữ liệu được gộp vào một cụm duy nhất ($K = 1$). Tuy nhiên, khi dữ liệu mở rộng, độ phức tạp tính toán của hệ thống chuyển dịch hiệu quả từ $O(N)$ sang $O(K + T)$, trong đó K và T là các hằng số được kiểm soát.

Kết quả này chứng minh tính khả thi của việc triển khai hệ thống trên các hạ tầng phần cứng phổ thông (như CPU Intel Core i5 dòng tiết kiệm điện) mà vẫn đảm bảo khả năng phục vụ hàng trăm người dùng đồng thời.

4.5 Kịch bản 4: Đánh giá Chatbot và Bảo mật PII

Kịch bản cuối cùng đánh giá khả năng của hệ thống EduTwin trong việc tích hợp mô hình ngôn ngữ lớn (LLM) để tạo ra trải nghiệm cá nhân hóa nhưng vẫn đảm bảo an toàn thông tin (Personalized & Secure Experience).

4.5.1 Kiểm thử PII Redaction (Bảo mật)

Mục tiêu của kiểm thử là đảm bảo các thông tin định danh cá nhân (PII) như Email, Số điện thoại và Tên riêng được tự động loại bỏ hoặc thay thế trước khi ngữ cảnh (context) được gửi sang LLM. Kết quả thực nghiệm tại Bảng 4.4 xác nhận module PII Redaction hoạt động chính xác, chuyển đổi các dữ liệu nhạy cảm thành các token ẩn danh (như <EMAIL_REDACTED>), qua đó giảm thiểu tối đa rủi ro rò rỉ dữ liệu người dùng.

4.5.2 Kiểm thử Cá nhân hóa với Context Injection

Kiểm thử này đánh giá tác động của việc thêm dữ liệu điểm số và hồ sơ người dùng vào Prompt để điều chỉnh hành vi của AI. Kết quả so sánh (Bảng 4.5) cho thấy việc bổ sung

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

Loại PII	Dữ liệu đầu vào	Dữ liệu sau xử lý
Email	nguyenvana@gmail.com<EMAIL_REDACTED>	
Số điện thoại	0912345678	<PHONE_REDACTED>
Tên riêng	Nguyễn Văn An	<NAME_REDACTED>

Bảng 4.4. Hiệu quả của module PII Redaction

ngữ cảnh đầy đủ (Mức 2) tuy làm tăng độ dài Prompt thêm 46% (từ 337 lên 494 tokens) nhưng đã thay đổi đáng kể chất lượng phản hồi. Thay vì chỉ phân tích số liệu khô khan, AI chuyển sang kết nối mục tiêu học tập với sở thích cá nhân, đồng thời thay đổi văn phong từ chuyên nghiệp sang thân thiện, năng động phù hợp với giới trẻ.

Tiêu chí	Mức 1: Chỉ có điểm số	Mức 2: Full Profile Cá nhân
Prompt Token	337 tokens	494 tokens (+46%)
Dữ liệu đầu vào	14 đầu điểm	Điểm số + 6 thuộc tính (Kinesthetic, Hướng ngoại, Mục tiêu Sư phạm, ...)
Chiến lược AI	Phân tích dựa trên số liệu: tập trung cải thiện môn yêu (Hóa) và duy trì môn mạnh (Toán).	Đề xuất dạy lại cho bạn bè (phù hợp định hướng Sư phạm), viết blog tiếng Anh (phù hợp sở thích).
Văn phong	Chuyên nghiệp, phân tích.	Vui vẻ, năng động, sử dụng emoji

Bảng 4.5. So sánh Prompt và Phản hồi giữa các mức độ Context

Về mặt định tính, hệ thống đã thể hiện khả năng thích ứng linh hoạt với từng *persona* người học. Trong kịch bản cơ bản (chỉ có điểm số), AI đề xuất lộ trình chuẩn 4 bước để cải thiện môn Hóa. Tuy nhiên, khi chuyển sang hồ sơ "Visual Learner" (người học qua thị giác), AI lập tức thay đổi chiến lược, đề xuất sử dụng Sơ đồ tư duy và Flashcards màu sắc. Đối với hồ sơ "Kinesthetic Learner" (người học qua vận động) có tính cách hướng ngoại, hệ thống lại gợi ý phương pháp Role-play và viết Blog, chứng tỏ độ nhạy ngữ cảnh (Context Sensitivity) cao. Dù prompt dài hơn, thời gian phản hồi vẫn đảm bảo trải nghiệm mượt mà.

4.6 Kịch bản 5: Đánh giá Learning Agent

Kịch bản này đánh giá khả năng suy luận nhiều bước của ReAct Agent trong chế độ Học tập (Learning Mode).

4.6.1 Kiểm thử Tool Usage

Đánh giá khả năng lựa chọn và sử dụng công cụ phù hợp:

Câu hỏi	Tool được chọn	Kết quả
"Tính $\sqrt{144} + 2^5$ "	Calculator	Kết quả chính xác: 44
"Ai là người sáng lập Apple?"	Wikipedia	Steve Jobs, Steve Wozniak
"Viết code Python tính tổng Python REPL 1-100"		Output: 5050
"Theo slide bài 3, công thức Document Search là gì?"		Trích dẫn từ tài liệu user

Bảng 4.6. Kết quả kiểm thử Tool Usage

4.6.2 Kiểm thử Multi-step Reasoning

Đánh giá khả năng suy luận nhiều bước với câu hỏi phức tạp:

Câu hỏi: "Nếu GDP Việt Nam năm 2023 là 430 tỷ USD và tăng trưởng 5%, GDP năm 2024 là bao nhiêu?"

Quá trình ReAct:

1. **Thought:** Ưu tiên tìm kiếm thông tin từ tài liệu người dùng.
2. **Action:** SearchUserDocuments("GDP Việt Nam 2023 tăng trưởng")
3. **Observation:** NO_USER_DOCUMENTS: Người dùng chưa tải lên tài liệu nào. Hãy tìm kiếm thông tin từ nguồn bên ngoài.

4. **Self-Evaluation:**

- Có đủ thông tin? No
- Độ chính xác: Low

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

- Thiếu gì: Cân tính toán GDP mới

5. **Thought:** Không có tài liệu người dùng. Đây là bài toán tính toán số học, cần sử dụng Calculator để đảm bảo độ chính xác. Công thức: $GDP_{2024} = GDP_{2023} \times (1 + \text{tỷ lệ tăng trưởng})$

6. **Action:** Calculator("430 * 1.05")

7. **Observation:** Result: 451.5

8. **Self-Evaluation:**

- Có đủ thông tin? Yes
- Độ chính xác: High (kết quả từ Calculator)
- Thiếu gì: Không

9. **Final Answer:** GDP Việt Nam năm 2024 ước tính là **451.5 tỷ USD**.

Cách tính: $430 \times 1.05 = 451.5$ (tỷ USD)

4.7 Bàn luận chung

Tổng hợp lại, chuỗi 5 kịch bản thực nghiệm đã chứng minh tính khả thi và hiệu quả của kiến trúc EduTwin trên năm khía cạnh chính.

Thứ nhất, về độ tin cậy của dữ liệu (Kịch bản 1):

Thuật toán KNN Imputation đã đóng vai trò nền tảng vững chắc khi khôi phục thành công cấu trúc tương quan của dữ liệu khuyết ($R^2 \approx 0.73$), vượt trội hoàn toàn so với các phương pháp thống kê truyền thống (Mean/Median chỉ đạt $R^2 \approx 0.17$). Ngay cả trong kịch bản cực đoan (50% dữ liệu bị thiếu), KNN vẫn duy trì $R^2 > 0.64$, chứng minh dữ liệu giáo dục có tính dư thừa thông tin cao và thuật toán đề xuất đã khai thác hiệu quả đặc tính này.

Thứ hai, về hiệu quả của Lazy Learning (Kịch bản 2):

Kết quả đã bác bỏ giả thuyết "mô hình càng phức tạp càng hiệu quả". KNN ($k = 15$) đạt độ chính xác cao nhất ($R^2 = 0.66$, MAE = 0.53) trong cả 5 mô hình, vượt qua cả LSTM và XGBoost. Điều này khẳng định dữ liệu điểm số học sinh mang tính chất cục bộ (local structure) mạnh mẽ, hoàn toàn tương thích với các thuật toán dựa trên láng giềng. Lazy Learning còn cung cấp khả năng giải thích cao (explainability) bằng việc truy xuất trực tiếp các học sinh "tương tự".

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

Thứ ba, về khả năng tối ưu hóa hiệu năng (Kịch bản 3):

Chiến lược Phân cụm K-Means kết hợp Tuyển chọn Mẫu hình Thích ứng đã chuyển đổi độ phức tạp từ $O(N)$ sang $O(K + T)$, đạt hệ số tăng tốc 11.5 lần tại quy mô 100,000 mẫu (từ 839ms xuống còn 73ms). Hệ thống loại bỏ hoàn toàn sự phụ thuộc vào GPU và hoạt động mượt mà trên phần cứng phổ thông, mở ra khả năng triển khai rộng rãi với chi phí thấp.

Thứ tư, về trải nghiệm cá nhân hóa và bảo mật (Kịch bản 4):

Hệ thống Chatbot đã chứng minh khả năng thích ứng với từng persona người học thông qua Context Injection. Việc bổ sung Full Profile (+46% tokens) đã thay đổi đáng kể chất lượng phản hồi: từ phân tích số liệu khô khan sang kết nối mục tiêu học tập với sở thích cá nhân. Song song đó, module PII Redaction đảm bảo an toàn tuyệt đối thông tin định danh trước khi gửi đến LLM.

Thứ năm, về khả năng suy luận của Learning Agent (Kịch bản 5):

ReAct Agent đã thể hiện năng lực suy luận nhiều bước và lựa chọn công cụ phù hợp cho từng loại câu hỏi: Calculator cho tính toán, Wikipedia cho tra cứu kiến thức, Python REPL cho lập trình, và Document Search cho tài liệu người dùng. Cơ chế này vượt xa khả năng của chatbot truyền thống, biến EduTwin thành một trợ lý học tập thông minh có khả năng giải quyết vấn đề phức tạp.

Kết luận:

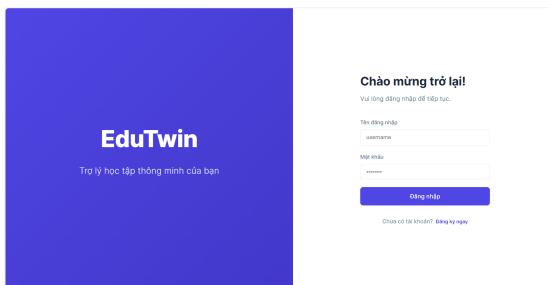
EduTwin đã thành công trong việc kết hợp ba trụ cột công nghệ: (1) Lazy Learning với tối ưu hóa K-Means cho dự đoán hiệu năng cao, (2) LLM với Context Injection và PII Redaction cho chatbot cá nhân hóa an toàn, và (3) ReAct Agent với Tool Use cho học tập chủ động. Kiến trúc này tạo nên một hệ sinh thái "Bản sao Học tập Kỹ thuật số" hoàn chỉnh, sẵn sàng phục vụ hàng trăm người dùng đồng thời với chi phí vận hành thấp.

4.8 Kết quả xây dựng ứng dụng

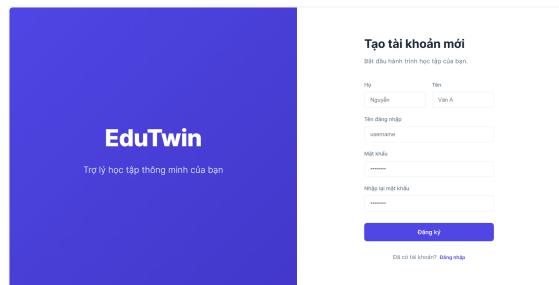
4.8.1 Giao diện đăng nhập/đăng ký

Giao diện xác thực người dùng được thiết kế theo phong cách hiện đại, tối giản, phân chia rõ ràng giữa khu vực nhận diện thương hiệu và khu vực nhập liệu. Để đảm bảo an toàn và trải nghiệm liền mạch, hệ thống tích hợp các cơ chế kiểm tra tính hợp lệ (validation) chặt chẽ ngay từ đầu vào đối với tên đăng nhập và mật khẩu. Người dùng sẽ được tự động chuyển hướng đến trang chatbot ngay sau khi quy trình xác thực thành công.

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



Hình 4.3. Giao diện đăng nhập



Hình 4.4. Giao diện đăng ký

4.8.2 Phân hệ Quản trị viên

Phân hệ quản trị cung cấp khả năng tùy biến sâu rộng thông qua tính năng định nghĩa cấu trúc giảng dạy. Quản trị viên có toàn quyền thiết lập số lượng mốc thời gian, danh sách môn học và loại thang điểm (thang 10, 100 hoặc Grade Point Average (GPA)) để phù hợp với đặc thù của từng hệ thống giáo dục. Ngoài ra, hệ thống cũng hỗ trợ các công cụ quản lý dữ liệu nền tảng, cho phép tải lên tập dữ liệu mẫu huấn luyện và cập nhật các tài liệu tham khảo chuyên sâu phục vụ cho module RAG.

Song song với quản lý cấu trúc, quản trị viên có thể can thiệp trực tiếp vào lõi công nghệ thông qua tính năng cấu hình mô hình Machine Learning. Hệ thống cho phép lựa chọn linh hoạt giữa các mô hình và tinh chỉnh các siêu tham số (hyperparameters) nhằm tối ưu hóa độ chính xác. Hiệu suất mô hình sau đó được đánh giá minh bạch thông qua quá trình kiểm thử với đầu vào/đầu ra động. Kết quả hiển thị trực quan qua các chỉ số thông kê chuẩn (MAE, RMSE, R²) và chỉ số Pseudo-Accuracy – một thước đo định lượng đơn giản hóa giúp người dùng không chuyên dễ dàng ra quyết định.

4.8.3 Phân hệ Người dùng

Điểm số và trực quan hóa dữ liệu Tại giao diện quản lý kết quả học tập, người dùng có thể nhập liệu trực tiếp trên lưới hoặc sử dụng tính năng Import/Export Excel. Nhờ công nghệ WebSocket, dữ liệu được đồng bộ thời gian thực với luồng xử lý Machine Learning, cho phép hệ thống hiển thị ngay lập tức các giá trị dự báo tương lai song song với điểm số thực tế. Dữ liệu này sau đó được trực quan hóa đa chiều qua các biểu đồ (đường, cột, radar), giúp người dùng dễ dàng theo dõi xu hướng. Đặc biệt, module phân tích AI (AI Insights) được tích hợp để tự động trích xuất thông tin ẩn, đưa ra nhận xét và đề xuất cải thiện dựa trên các mẫu hình dữ liệu ghi nhận được.

Giao diện Trò chuyện (Chat Mode)

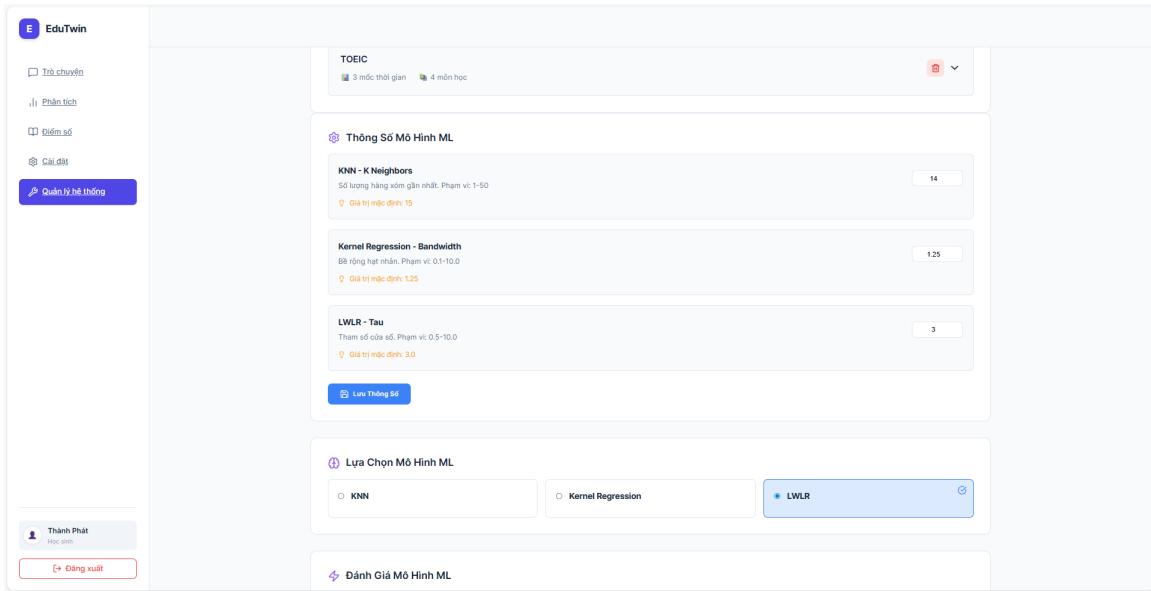
Trợ lý ảo EduTwin sở hữu giao diện thân thiện với khả năng phản hồi thời gian thực qua

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

Hình 4.5. Tính năng cho phép quản trị viên (Admin) định nghĩa cấu trúc giảng dạy

Hình 4.6. Giao diện cài đặt cho các cấu trúc giảng dạy đã thiết lập

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



Hình 4.7. Các tính năng cho phép cài đặt hệ thống ML như lựa chọn mô hình, lựa chọn siêu tham số

WebSocket và hiển thị nội dung đa dạng qua định dạng Markdown. Hệ thống tự động nạp ngũ cảnh điểm số và lịch sử hội thoại (20 tin nhắn gần nhất) để các câu trả lời luôn bám sát thực tế học tập. Đặc biệt, module Hybrid Personalization liên tục học hỏi sở thích người dùng qua Keyword Detection và LLM Analysis, từ đó tinh chỉnh phong cách tương tác phù hợp với từng học sinh.

Giao diện Học tập (Learning Mode)

Chế độ Learning Agent được thiết kế cho các câu hỏi học tập phức tạp, yêu cầu suy luận nhiều bước. Giao diện hiển thị trực quan quá trình ReAct (Reasoning + Acting) của Agent: từng bước Thought, Action và Observation được cập nhật real-time qua WebSocket, giúp người dùng theo dõi "cách nghĩ" của AI. Agent được trang bị 4 công cụ: Calculator (tính toán), Wikipedia (tra cứu), Python REPL (lập trình), và Document Search (tìm kiếm trong tài liệu đã upload). Người dùng có thể tải lên tài liệu học tập (PDF, DOCX, TXT) để Agent trích dẫn khi trả lời.

Giao diện Cài đặt

Màn hình cài đặt cho phép người dùng quản lý thông tin cá nhân và xem các sở thích đã được hệ thống học tự động (Learned Preferences). Các thuộc tính như phong cách học tập (Visual, Auditory, Kinesthetic), tính cách (Introvert, Extrovert), phong cách giao tiếp, sở thích và mục tiêu học tập được hiển thị rõ ràng, giúp người dùng hiểu cách AI "nhìn nhận" về mình và có thể điều chỉnh nếu cần.

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

⚡ Đánh Giá Mô Hình ML

Chọn mốc thời gian để đánh giá và so sánh các mô hình

📊 Đầu vào (chọn nhiều):

- Học kỳ 1 lớp 11
- Học kỳ 2 lớp 11
- Học kỳ 1 lớp 12
- Học kỳ 2 lớp 12

✓ Đã chọn: 4 mốc

⌚ Mục tiêu dự đoán (chọn nhiều):

- Học kỳ 1 lớp 11 (Không hợp lệ)
- Học kỳ 2 lớp 11 (Không hợp lệ)
- Học kỳ 1 lớp 12
- Học kỳ 2 lớp 12

✓ Đã chọn: 2 mốc

⚡ Đang đánh giá mô hình...

Hình 4.8. Tính năng đánh giá mô hình tùy biến với input features và output labels thay đổi theo cấu trúc giảng dạy được thiết lập

⚡ Đánh Giá Mô Hình ML

Chọn mốc thời gian để đánh giá và so sánh các mô hình

📊 Đầu vào (chọn nhiều):

- Học kỳ 1 lớp 11
- Học kỳ 2 lớp 11
- Học kỳ 1 lớp 12
- Học kỳ 2 lớp 12

✓ Đã chọn: 4 mốc

⌚ Mục tiêu dự đoán (chọn nhiều):

- Học kỳ 1 lớp 11 (Không hợp lệ)
- Học kỳ 2 lớp 11 (Không hợp lệ)
- Học kỳ 1 lớp 12
- Học kỳ 2 lớp 12

✓ Đã chọn: 2 mốc

⚡ Đánh Giá Mô Hình

✓ Đánh giá hoàn tất!

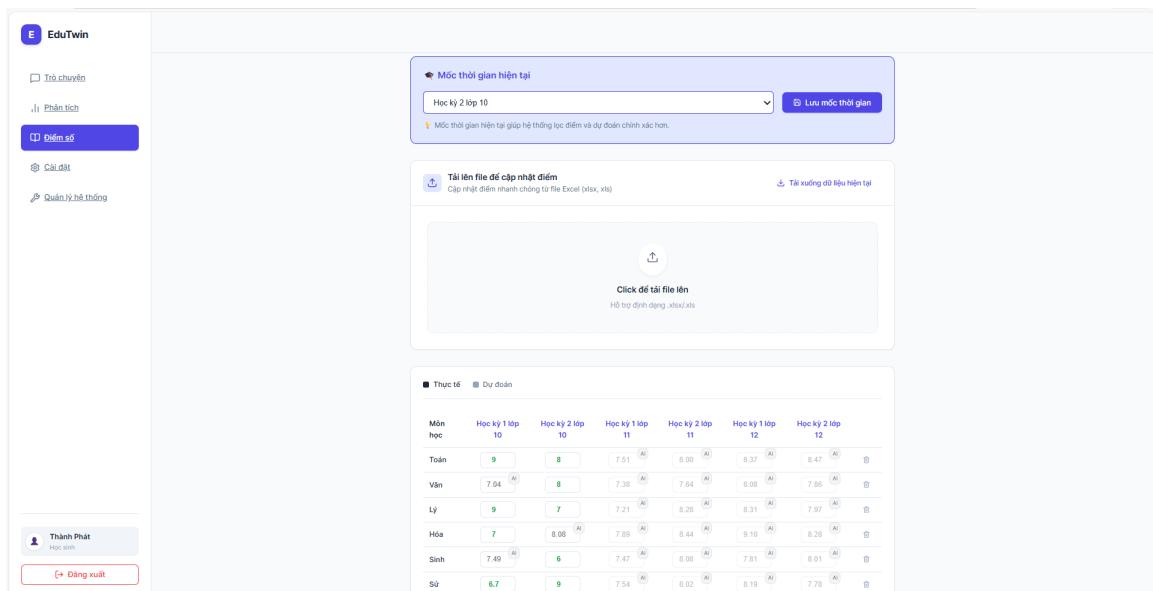
Cấu trúc: THPT
Dataset: 2283 mẫu (Train: 1826, Test: 457)

⌚ Mô hình được đề xuất:
KNN
Độ chính xác: **97.75%**

Mô hình	MAE	MSE	RMSE	Độ chính xác
KNN	0.2253	0.0848	0.2911	97.75%
Kernel Regression	0.2286	0.0878	0.2963	97.71%
LWLR	0.2274	0.0861	0.2934	97.73%

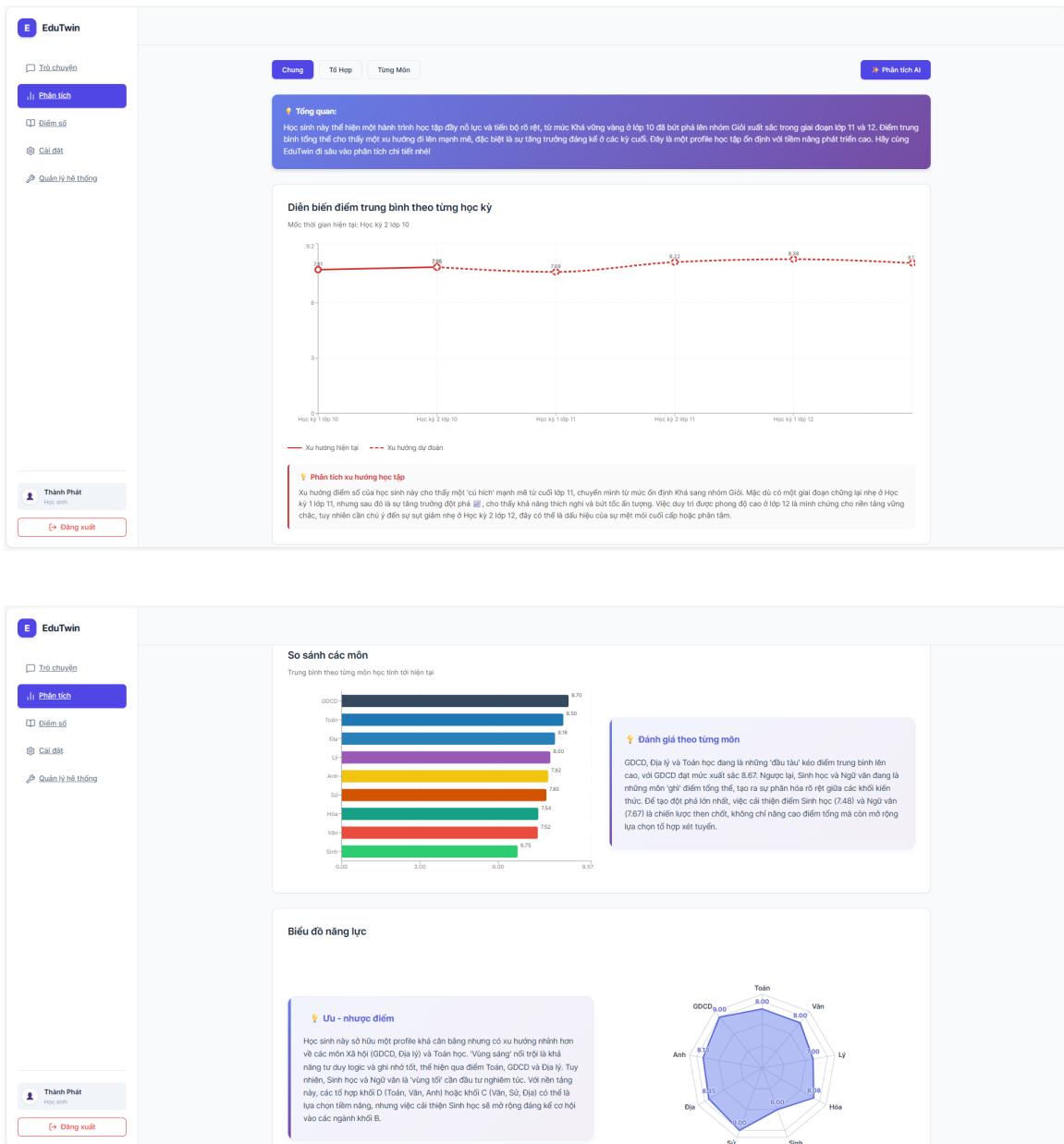
Hình 4.9. Kết quả đánh giá mô hình trực quan với đầy đủ tham số và đề xuất phù hợp

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



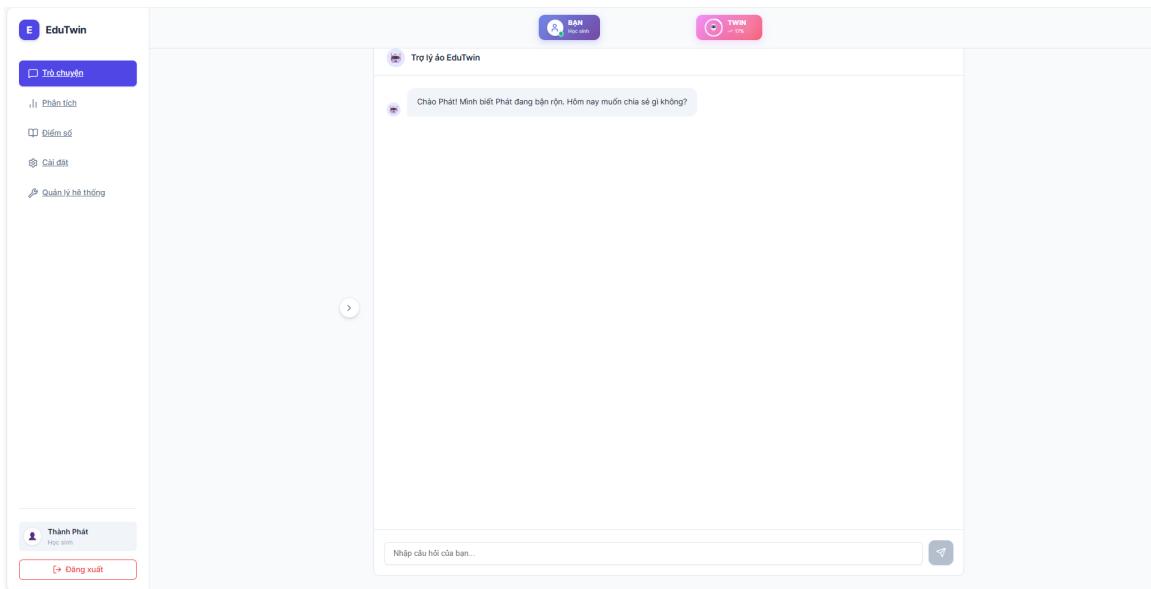
Hình 4.10. Giao diện tổng hợp và quản lý kết quả học tập hiện tại của học sinh cũng như các dự đoán của hệ thống

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ

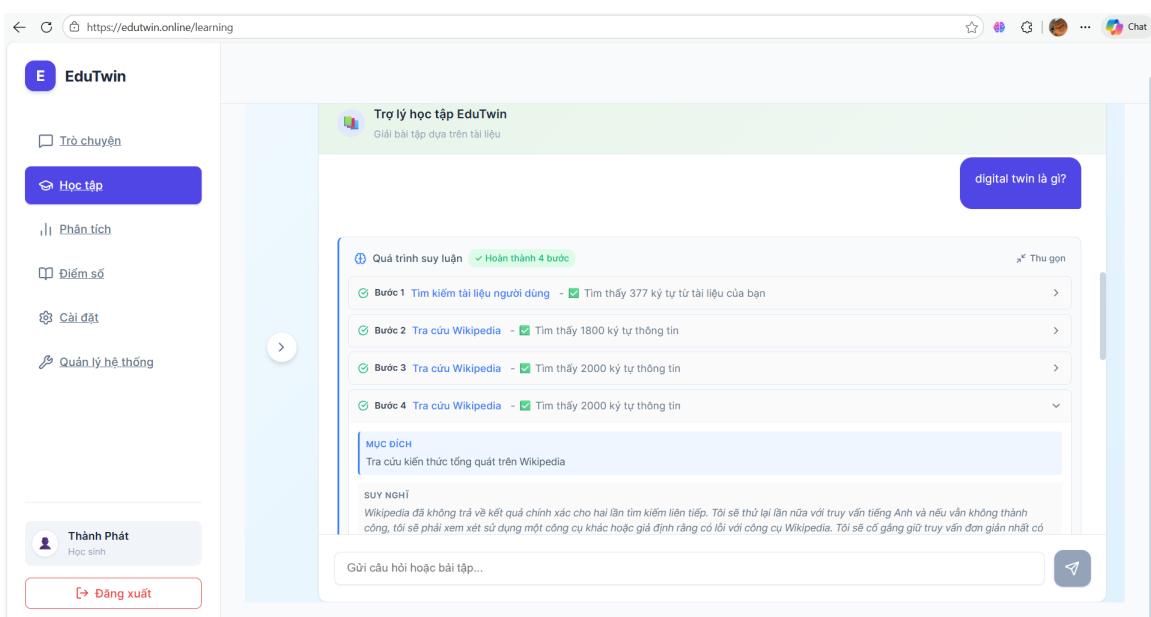


Hình 4.11. Giao diện biểu đồ trực quan hóa với phân tích từ AI giúp người dùng khai thác thông tin và nắm bắt các thông tin ẩn của dữ liệu

CHƯƠNG 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ



Hình 4.12. Giao diện Chatbot với phản hồi real-time và hỗ trợ Markdown



Hình 4.13. Giao diện Learning Agent với hiển thị tiến trình suy luận ReAct



Hình 4.14. Giao diện Cài đặt với danh sách sở thích đã học (Learned Preferences)

Chương 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1.1 Kết luận chung

Khóa luận đã nghiên cứu và xây dựng thành công hệ thống *EduTwin – Bản sao Học tập Kỹ thuật số tích hợp AI*, giải quyết trọn vẹn bài toán cá nhân hóa giáo dục và dự báo hiệu suất học tập trong thời gian thực. Thông qua quá trình nghiên cứu lý thuyết, thiết kế kiến trúc và thực nghiệm trên bộ dữ liệu thực tế, khóa luận đã đạt được những kết quả quan trọng sau:

Chứng minh hiệu quả vượt trội của Lazy Learning trong giáo dục:

Kết quả thực nghiệm đã bác bỏ quan điểm cho rằng mô hình càng phức tạp (Deep Learning) thì càng chính xác. Với đặc thù dữ liệu điểm số học sinh mang tính cấu trúc cục bộ và tương quan láng giềng mạnh mẽ, thuật toán KNN với $k = 15$ đã đạt độ chính xác cao nhất ($R^2 = 0.66$, $MAE = 0.53$), vượt qua cả LSTM và XGBoost, đồng thời có chi phí tính toán thấp hơn đáng kể.

Giải quyết bài toán độ trễ và khả năng mở rộng:

Bằng cách áp dụng kiến trúc lai ghép với kỹ thuật Phân cụm K-Means và Tuyển chọn Mẫu hình Thích ứng (Cluster Indexing & Adaptive Prototype Selection), hệ thống đã khắc phục nhược điểm về tốc độ của Lazy Learning. Kết quả Stress Test cho thấy hệ thống đạt tốc độ tăng tốc (speedup) lên tới 11.5 lần ở quy mô 100.000 bản ghi, giảm độ trễ từ 839ms xuống còn 73ms, đảm bảo khả năng triển khai thực tế trên hạ tầng phần cứng phổ thông.

Kiến trúc linh hoạt hướng Siêu dữ liệu (Metadata-Driven):

EduTwin không bị giới hạn bởi một chương trình học cố định nhờ thiết kế cơ sở dữ liệu linh hoạt sử dụng PostgreSQL JSONB. Hệ thống cho phép các đơn vị giáo dục tự định nghĩa cấu trúc môn học, thang điểm và quy chế đánh giá mà không cần can thiệp vào mã nguồn hay huấn luyện lại mô hình AI.

Trải nghiệm tương tác thông minh và bảo mật:

Việc tích hợp LLM (Google Gemini) với kỹ thuật Context Injection và cơ chế PII Redaction đã biến hệ thống từ một công cụ thông kê thành một trợ lý ảo thấu hiểu ngữ cảnh. EduTwin không chỉ bảo vệ an toàn thông tin định danh của người học mà còn đưa ra các tư vấn sư phạm được cá nhân hóa sâu sắc dựa trên hồ sơ năng lực và phong cách học tập (Visual/Kinesthetic). Module Hybrid Personalization liên tục học sở thích người dùng qua Keyword Detection và LLM Analysis.

Learning Agent với kiến trúc ReAct:

Ngoài Chatbot truyền thống, hệ thống còn tích hợp Learning Agent sử dụng mẫu thiết kế ReAct (Reasoning + Acting), cho phép suy luận nhiều bước với 4 công cụ tích hợp (Calculator, Wikipedia, Python REPL, Document Search). Cơ chế Self-Reflection giúp giảm thiểu hallucination và tăng độ tin cậy của phản hồi trong ngữ cảnh học tập.

5.1.2 Các hạn chế của đề tài

Bên cạnh những kết quả đạt được, khóa luận vẫn còn một số hạn chế cần được nhìn nhận:

Phạm vi dữ liệu:

Hiện tại, hệ thống mới chỉ được kiểm chứng trên dữ liệu điểm số định lượng (Structured Grades). Các dữ liệu phi cấu trúc quan trọng khác như lịch sử điểm danh, hành vi click chuột trên LMS hay các hoạt động ngoại khóa chưa được đưa vào mô hình dự báo.

Giới hạn của thuật toán KNN:

Mặc dù hoạt động rất tốt trong việc nội suy (interpolation) với dữ liệu phân phôi dày, KNN có thể gặp khó khăn trong việc ngoại suy (extrapolation) đối với các trường hợp học sinh có hồ sơ năng lực quá đặc biệt hoặc nằm ngoài vùng phủ của dữ liệu huấn luyện (outliers).

Phụ thuộc vào API bên thứ ba:

Chức năng Chatbot và Learning Agent hiện tại phụ thuộc vào API của Google Gemini. Điều này có thể dẫn đến rủi ro về chi phí vận hành tăng cao khi quy mô người dùng lớn, cũng như vấn đề về độ trễ khi API quá tải.

5.1.3 Hướng phát triển

Để hoàn thiện và nâng cao khả năng ứng dụng của EduTwin trong thực tiễn, hướng phát triển trong tương lai sẽ tập trung vào các nội dung sau:

Mở rộng nguồn dữ liệu và Mô hình Hybrid:

Nghiên cứu tích hợp thêm các nguồn dữ liệu hành vi (Behavioral Data) từ các hệ thống LMS. Đồng thời, xem xét phát triển mô hình lai (Hybrid Model) kết hợp giữa Lazy Learning (cho dự báo ngắn hạn, cục bộ) và Deep Learning (cho dự báo xu hướng dài hạn) để tận dụng ưu điểm của cả hai phương pháp.

Tối ưu hóa chi phí LLM (Small Language Models):

Nghiên cứu tinh chỉnh (Fine-tuning) các mô hình ngôn ngữ nhỏ hơn (SLM) hoặc các mô hình mã nguồn mở (như Llama, Gemma) để có thể tự triển khai (Self-hosted) ngay trên máy chủ của trường học. Điều này giúp giảm chi phí API, tăng tốc độ phản hồi và đảm bảo

CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

quyền riêng tư dữ liệu tuyệt đối.

Phát triển Hệ sinh thái Đa nền tảng:

Mở rộng EduTwin từ ứng dụng Web sang ứng dụng di động (Mobile App) để tăng tính tiện lợi cho học sinh. Đồng thời, xây dựng thêm phân hệ dành cho Phụ huynh và Giáo viên, tạo thành một vòng tròn khép kín giữa Gia đình – Nhà trường – Học sinh, giúp việc giám sát và hỗ trợ người học trở nên đồng bộ và hiệu quả hơn.

Cơ chế Tự động hóa Tham số (Auto-Tuning):

Hiện tại các tham số như k (số lảng giềng), h (bandwidth) hay τ (LWLR) đang được chọn dựa trên thực nghiệm hoặc cấu hình thủ công. Hướng phát triển tiếp theo là xây dựng cơ chế AutoML để hệ thống tự động học và điều chỉnh các tham số này theo thời gian thực dựa trên sự biến đổi của dữ liệu mới nạp vào.

Nâng cao Learning Agent:

Mở rộng bộ công cụ (Tools) của Learning Agent, bao gồm: tích hợp tìm kiếm web real-time, hỗ trợ vẽ biểu đồ trực quan, và kết nối với các API giáo dục bên ngoài (như Wolfram Alpha). Nghiên cứu cải tiến cơ chế Self-Reflection để tăng độ chính xác và giảm thời gian suy luận.

Phụ lục A. CÁC THÔNG TIN LIÊN QUAN

A.1 Tài nguyên thực nghiệm

Các tài nguyên phục vụ cho quá trình thực nghiệm của hệ thống EduTwin được liệt kê như sau:

- **Tập dữ liệu thực nghiệm:** [Google Sheets Dataset](#)
- **Notebook huấn luyện và đánh giá mô hình:** [Training & Evaluation Notebook](#)
- **Thông tin chi tiết Prompt và Response (Kịch bản 4):** [LLM Prompt & Response Documentation](#)

TÀI LIỆU THAM KHẢO

- [1] Thủ tướng Chính phủ. *Quyết định số 131/QĐ-TTg: Phê duyệt Đề án "Tăng cường ứng dụng công nghệ thông tin và chuyển đổi số trong giáo dục và đào tạo giai đoạn 2022–2025, định hướng đến năm 2030".* <https://vanban.chinhphu.vn/?pageid=27160&docid=205236&classid=0>. Người ký: Vũ Đức Đam. Jan. 2022.
- [2] Hossein Omrany, K. Al-Obaidi, Amirhosein Ghaffarianhoseini, Rui-Dong Chang, Chansik Park, và F. Rahimian. “Digital twin technology for education, training and learning in construction industry: implications for research and practice”. In: *Engineering, Construction and Architectural Management* (2025). DOI: [10.1108/ecam-10-2024-1376](https://doi.org/10.1108/ecam-10-2024-1376).
- [3] Jie Zhang, Jingdong Zhu, Weiwei Tu, Minkai Wang, Yiling Yang, Fang Qian, và Yeqing Xu. “The Effectiveness of a Digital Twin Learning System in Assisting Engineering Education Courses: A Case of Landscape Architecture”. In: *Applied Sciences* (2024). DOI: [10.3390/app14156484](https://doi.org/10.3390/app14156484).
- [4] L. Rovati, Phillip J. Gary, E. Cubro, Yue Dong, O. Kilickaya, Phillip J. Schulte, Xiang Zhong, M. Wörster, D. Kelm, O. Gajic, Alexander S. Niven, và Amos Lal. “Development and usability testing of a patient digital twin for critical care education: a mixed methods study”. In: *Frontiers in Medicine* 10 (2024). DOI: [10.3389/fmed.2023.1336897](https://doi.org/10.3389/fmed.2023.1336897).
- [5] Yuwei Tao và Aijuan Xie. “Research on the architecture and practice of inquiry learning model integrated with digital twin”. In: *Interactive Learning Environments* (2025). DOI: [10.1080/10494820.2025.2476716](https://doi.org/10.1080/10494820.2025.2476716).
- [6] Jean Baptiste Habarurema, Raffaele Di Fuccio, và P. Limone. “Enhancing e-learning with a digital twin for innovative learning”. In: *The International Journal of Information and Learning Technology* (2025). DOI: [10.1108/ijilt-02-2024-0034](https://doi.org/10.1108/ijilt-02-2024-0034).
- [7] W. Villegas-Ch., Diego Buenaño-Fernández, Alexandra Maldonado Navarro, và Aracely Mera-Navarrete. “Adaptive intelligent tutoring systems for STEM education: analysis of the learning impact and effectiveness of

- personalized feedback”. In: *Smart Learn. Environ.* 12 (2025), p. 41. DOI: [10.1186/s40561-025-00389-y](https://doi.org/10.1186/s40561-025-00389-y).
- [8] Meltem Taşkın. “Artificial Intelligence in Personalized Education: Enhancing Learning Outcomes Through Adaptive Technologies and Data-Driven Insights”. In: *Human Computer Interaction* (2025). DOI: [10.62802/ygye0506](https://doi.org/10.62802/ygye0506).
- [9] Ani Grubišić, Ines Šarić-Grgić, A. Gašpar, và Branko Žitko. “Usability Evaluation of an Adaptive Courseware Approach in the Natural Language-Based Intelligent Tutoring System-Tutomat”. In: *Journal of Computer Assisted Learning* (2025). DOI: [10.1111/jcal.70071](https://doi.org/10.1111/jcal.70071).
- [10] Ramesh Singh, Chenlep Yakha Konyak, và Akangjungshi Longkumer. “A Multi-Access Edge Computing Approach to Intelligent Tutoring Systems for Real-Time Adaptive Learning”. In: *International Journal of Information Technology* (2025). DOI: [10.1007/s41870-025-02460-w](https://doi.org/10.1007/s41870-025-02460-w).
- [11] Shahzad Rizwan, Chee Ken Nee, và Salem Garfan. “Identifying the Factors Affecting Student Academic Performance and Engagement Prediction in MOOC Using Deep Learning: A Systematic Literature Review”. In: *IEEE Access* 13 (2025), pp. 18952–18982. DOI: [10.1109/access.2025.3533915](https://doi.org/10.1109/access.2025.3533915).
- [12] Jialun Pan, Zhanzhan Zhao, và Dongkun Han. “Academic Performance Prediction Using Machine Learning Approaches: A Survey”. In: *IEEE Transactions on Learning Technologies* 18 (2025), pp. 351–368. DOI: [10.1109/tlt.2025.3554174](https://doi.org/10.1109/tlt.2025.3554174).
- [13] Qi Lang, Minjuan Wang, Minghao Yin, Shuang Liang, và Wenzhuo Song. “Transforming Education With Generative AI (GAI): Key Insights and Future Prospects”. In: *IEEE Transactions on Learning Technologies* 18 (2025), pp. 230–242. DOI: [10.1109/tlt.2025.3537618](https://doi.org/10.1109/tlt.2025.3537618).
- [14] Michail N. Giannakos, Roger Azevedo, Peter Brusilovsky, M. Cukurova, Y. Dimitriadis, Davinia Hernández Leo, Sanna Järvelä, M. Mavrikis, và Bart Rienties. “The promise and challenges of generative AI in education”. In: *Behav. Inf. Technol.* 44 (2024), pp. 2518–2544. DOI: [10.1080/0144929x.2024.2394886](https://doi.org/10.1080/0144929x.2024.2394886).
- [15] V. Nikolovski, D. Trajanov, và Ivan Chorbev. “Advancing AI in Higher Education: A Comparative Study of Large Language Model-Based Agents for Exam Question Generation, Improvement, and Evaluation”. In: *Algorithms* (2025). DOI: [10.3390/a18030144](https://doi.org/10.3390/a18030144).

TÀI LIỆU THAM KHẢO

- [16] M. B. Jelodar. “Generative AI, Large Language Models, and ChatGPT in Construction Education, Training, and Practice”. In: *Buildings* (2025). DOI: [10.3390/buildings15060933](https://doi.org/10.3390/buildings15060933).
- [17] Xiaojun Xu, Yixiao Chen, và Jing Miao. “Opportunities, challenges, and future directions of large language models, including ChatGPT in medical education: a systematic scoping review”. In: *Journal of Educational Evaluation for Health Professions* 21 (2024). DOI: [10.3352/jeehp.2024.21.6](https://doi.org/10.3352/jeehp.2024.21.6).
- [18] Zachary K. Collier, Kamal Chawla, và Olushola O. Soyoye. “Optimizing Imputation for Educational Data: Exploring Training Partition and Missing Data Ratios”. In: *The Journal of Experimental Education* 93 (2024), pp. 607–627. DOI: [10.1080/00220973.2023.2287447](https://doi.org/10.1080/00220973.2023.2287447).
- [19] Gabriel-Vasilică Sasu, Bogdan-Iulian Ciubotaru, Nicolae Goga, và A. Vasilăteanu. “Addressing Missing Data Challenges in Geriatric Health Monitoring: A Study of Statistical and Machine Learning Imputation Methods”. In: *Sensors (Basel, Switzerland)* 25 (2025). DOI: [10.3390/s25030614](https://doi.org/10.3390/s25030614).
- [20] Abdulaziz Altamimi, Aisha Ahmed Alarfaj, Muhammad Umer, E. Alabdulqader, Shtwai Alsubai, Tai-hoon Kim, và Imran Ashraf. “An automated approach to predict diabetic patients using KNN imputation and effective data mining techniques”. In: *BMC Medical Research Methodology* 24 (2024). DOI: [10.1186/s12874-024-02324-0](https://doi.org/10.1186/s12874-024-02324-0).
- [21] Khaled Alnowaiser. “Improving Healthcare Prediction of Diabetic Patients Using KNN Imputed Features and Tri-Ensemble Model”. In: *IEEE Access* 12 (2024), pp. 16783–16793. DOI: [10.1109/access.2024.3359760](https://doi.org/10.1109/access.2024.3359760).
- [22] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, và Yuan Cao. “React: Synergizing reasoning and acting in language models”. In: *The eleventh international conference on learning representations*. 2022.