

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



Hoàng Minh Nhật

**HỆ THỐNG GIỚI THIỆU KẾT BẠN DỰA
TRÊN TÍNH CÁCH, XỬ LÝ TRỰC TIẾP
TRÊN THIẾT BỊ ĐỂ BẢO VỆ QUYỀN
RIÊNG TƯ**

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC

Ngành: Công nghệ thông tin

TP. HCM - 2025

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**



Hoàng Minh Nhật

**HỆ THỐNG GIỚI THIỆU KẾT BẠN DỰA
TRÊN TÍNH CÁCH, XỬ LÝ TRỰC TIẾP
TRÊN THIẾT BỊ ĐỂ BẢO VỆ QUYỀN
RIÊNG TƯ**

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC

Ngành: Công nghệ thông tin

Cán bộ hướng dẫn: Th.S Ngô Khánh Khoa

TP. HCM - 2025

LỜI CAM ĐOAN

Khóa luận tốt nghiệp này được thực hiện trực tiếp bởi em và dưới sự hướng dẫn của thầy Thạc Sĩ Ngô Khánh Khoa. Em xin cam đoan rằng mọi quá trình nghiên cứu, phát triển, triển khai và báo cáo được trình bày trong báo cáo này đều được chính em độc lập thực hiện mà không sao chép, đạo văn từ những nguồn khác mà không có sự cho phép. Nếu có vi phạm quy định về nội dung trí tuệ, em xin chịu trách nhiệm tất cả những truy cứu theo quy định của Trường Đại Học Công Nghệ Thông Tin - ĐHQGHCM.

TpHCM, ngày 26 tháng 12 năm 2024

Sinh viên

Hoàng Minh Nhật

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành đến Ban giám hiệu Trường Đại học Công nghệ Thông tin – ĐHQG-HCM cùng các Thầy Cô trong Khoa đã tạo điều kiện và truyền đạt những kiến thức quý báu trong suốt quá trình em học tập tại trường.

Đặc biệt, em xin bày tỏ lòng tri ân sâu sắc nhất đến thầy Ngô Khánh Khoa. Trong suốt thời gian thực hiện đề tài, thầy không chỉ hỗ trợ tận tình về mặt kiến thức chuyên môn mà còn là một người thầy tâm huyết, luôn dành thời gian sát cánh và đưa ra những lời khuyên sâu sắc giúp em hoàn thành tốt khóa luận này.

Hoàng Minh Nhật

TÓM TẮT

Khóa luận tập trung giải quyết bài toán xây dựng hệ thống giới thiệu kết bạn dựa trên sự tương đồng về tính cách, đồng thời đảm bảo quyền riêng tư cho dữ liệu người dùng thông qua mô hình xử lý trực tiếp trên thiết bị (on-device processing). Trong bối cảnh các mạng xã hội hiện đại đang đối mặt với nhiều thách thức về bảo mật thông tin nhạy cảm, đề tài đề xuất một quy trình hoàn chỉnh từ thu thập dữ liệu trả lời câu hỏi của mô hình tính cách Big-5, chuyển đổi sang không gian vector thu gọn bằng Phân tích Thành phần chính (PCA-4), cho đến việc lưu trữ an toàn bằng chuẩn mã hóa AES-256-GCM thông qua các hàm thực thi biên (Edge Functions). Hệ thống sử dụng thuật toán giới thiệu lai kết hợp giữa độ tương đồng tính cách, hành vi xã giao qua điểm số ELO và nhúng ngữ nghĩa (semantic embedding) cho sở thích cá nhân. Kết quả thực nghiệm trên hệ thống thật cho thấy mô hình PCA-4 giữ được hơn 90% phương sai dữ liệu gốc, cho phép xác định các cặp người dùng tương đồng với độ chính xác cao (0.9999). Hiệu năng hệ thống được tối ưu hóa đạt mức trễ chấp nhận được với trung bình 1.8 giây cho quy trình đăng ký và 2.3 giây cho việc tạo danh sách giới thiệu, đồng thời khẳng định tính khả thi của kiến trúc “Quyền riêng tư theo thiết kế” trong việc bảo vệ dữ liệu nhạy cảm mà vẫn duy trì hiệu quả kết nối cộng đồng.

Từ khóa: Twins, Big-5, PCA, Quyền riêng tư, Hệ thống giới thiệu, Mã hóa dữ liệu

MỤC LỤC

| | |
|------------------------------------------------------------|----------|
| Lời cam đoan | i |
| Lời cảm ơn | ii |
| Tóm tắt | iii |
| Mục lục | iv |
| Danh mục hình ảnh | x |
| Danh mục bảng | xi |
| Danh mục giải thuật | xii |
| Chương 1. Giới thiệu | 1 |
| 1.1. Bối cảnh và vấn đề | 1 |
| 1.1.1. Mạng xã hội và nhu cầu kết nối theo tính cách | 1 |
| 1.1.2. Rủi ro dữ liệu tính cách và yêu cầu bảo vệ | 2 |
| 1.2. Mục tiêu và phạm vi | 3 |
| 1.2.1. Mục tiêu chính | 3 |
| 1.2.2. Phạm vi thực hiện | 4 |
| 1.3. Bài toán và cách tiếp cận | 4 |
| 1.3.1. Bài toán chuyển đổi dữ liệu tính cách | 4 |
| 1.3.2. Bài toán bảo mật dữ liệu | 5 |
| 1.4. Đóng góp chính | 6 |
| 1.4.1. Đóng góp về mô hình chuyển đổi | 6 |
| 1.4.2. Đóng góp về bảo mật | 6 |
| 1.4.3. Đóng góp về tài liệu kỹ thuật và minh chứng | 6 |
| 1.5. Cấu trúc của báo cáo | 7 |
| Chương 2. Tổng quan quy trình hệ thống | 8 |

| | |
|----------------------------------------------------------------|-----------|
| 2.1. Mục tiêu của chương | 8 |
| 2.2. Các nguồn dữ liệu đầu vào | 8 |
| 2.2.1. Bộ câu hỏi Big Five và cách lấy mẫu | 8 |
| 2.2.2. Dữ liệu khảo sát công khai cho PCA | 9 |
| 2.2.3. Dữ liệu sở thích (hobbies) | 9 |
| 2.3. Tổng quan quy trình và tác nhân | 9 |
| 2.4. Đề xuất phân mảnh địa lý trong quy trình giới thiệu | 10 |
| 2.5. Mô hình giới thiệu và trọng số trong giới thiệu | 11 |
| 2.5.1. Điểm tương đồng tính cách (PCA) | 11 |
| 2.5.2. ELO từ tương tác like/skip | 11 |
| 2.5.3. Embedding sở thích và cosine similarity | 12 |
| 2.5.4. Trọng số tổng hợp | 12 |
| 2.6. Luồng dữ liệu chi tiết theo tác nhân | 13 |
| 2.6.1. Thiết bị người dùng | 13 |
| 2.6.2. Edge Function | 13 |
| 2.6.3. Cơ sở dữ liệu | 13 |
| 2.7. Cấu trúc các chương tiếp theo | 14 |
| Chương 3. Chuyển đổi dữ liệu tính cách (PCA-4) | 15 |
| 3.1. Mục tiêu của chương | 15 |
| 3.2. Big Five trong bối cảnh các mô hình tính cách | 15 |
| 3.2.1. Mô hình Chỉ báo Phân loại Myers-Briggs (MBTI) | 15 |
| 3.2.2. Mô hình tính cách HEXACO | 16 |
| 3.3. Chuẩn hóa điểm Big Five | 17 |
| 3.3.1. Thang đo và hướng câu hỏi | 17 |

| | |
|------------------------------------------------------------------|-----------|
| 3.3.2. Ví dụ định dạng dữ liệu đầu vào | 18 |
| 3.3.3. Vì sao chọn PCA-4 sau khi chuẩn hóa | 18 |
| 3.4. Đề xuất PCA-4 | 18 |
| 3.5. Huấn luyện PCA | 19 |
| 3.5.1. Nguồn dữ liệu và quy mô | 19 |
| 3.5.2. Công thức chiêu PCA | 20 |
| 3.6. Triển khai PCA trên thiết bị | 20 |
| 3.6.1. Cách triển khai | 20 |
| 3.6.2. Định dạng lưu trữ | 21 |
| 3.7. Thảo luận lựa chọn PCA | 21 |
| Chương 4. Bảo mật và mã hóa dữ liệu | 22 |
| 4.1. Mục tiêu của chương | 22 |
| 4.2. Tổng quan về cơ chế AES-GCM | 22 |
| 4.2.1. Nguyên lý cơ bản | 22 |
| 4.2.2. Đầu vào và đầu ra của AES-GCM | 23 |
| 4.3. Dữ liệu đầu vào từ góc nhìn người dùng | 24 |
| 4.3.1. Trải nghiệm nhập liệu và ranh giới dữ liệu nhạy cảm | 24 |
| 4.3.2. Chuyển đổi trên thiết bị | 24 |
| 4.4. Mã hóa dữ liệu bằng AES-256-GCM | 25 |
| 4.4.1. Đề xuất AES-GCM | 25 |
| 4.4.2. Lý do chọn AES-GCM | 25 |
| 4.4.3. Lựa chọn thay thế: RSA | 25 |
| 4.4.4. Lựa chọn thay thế: Bcrypt/Scrypt | 26 |
| 4.4.5. Lựa chọn thay thế: Homomorphic encryption | 27 |

| | |
|---------------------------------------------------------|-----------|
| 4.4.6. Lựa chọn thay thế: Differential privacy | 28 |
| 4.4.7. Vai trò của Edge Function và khóa bí mật | 28 |
| 4.4.8. Lưu trữ và giới hạn truy cập | 29 |
| 4.5. Dữ liệu sở thích và mã hóa | 29 |
| Chương 5. Hệ giới thiệu và cơ chế xếp hạng | 31 |
| 5.1. Mục tiêu của chương | 31 |
| 5.2. Vì sao vẫn cần tính cách khi đã có sở thích | 31 |
| 5.3. Đề xuất thuật toán ELO | 32 |
| 5.3.1. Vai trò của ELO trong hành vi xã giao | 32 |
| 5.3.2. Bàn luận về thiết kế ELO | 33 |
| 5.4. Nguồn sử dụng sở thích | 33 |
| 5.5. Đề xuất mô hình ngữ nghĩa (semantic model) | 33 |
| 5.5.1. Lựa chọn thay thế: TF-IDF | 35 |
| 5.5.2. Lựa chọn thay thế: Word2Vec | 35 |
| 5.6. Công thức xếp hạng tổng hợp | 36 |
| 5.6.1. Bàn luận về trọng số | 36 |
| 5.6.2. Ví dụ minh họa xếp hạng | 37 |
| 5.7. Bảo vệ dữ liệu sở thích và quyền riêng tư | 38 |
| Chương 6. Thực nghiệm và Đánh giá | 39 |
| 6.1. Mục tiêu của chương | 39 |
| 6.2. Câu hỏi nghiên cứu (Research Questions) | 39 |
| 6.3. Thiết lập thực nghiệm | 40 |
| 6.3.1. Môi trường và công cụ | 40 |
| 6.3.2. Tập dữ liệu | 40 |

| | |
|---------------------------------------------------------------------|-----------|
| 6.4. Kết quả và phân tích | 41 |
| 6.4.1. RQ1: Hiệu quả của mô hình PCA-4 và độ tương đồng cosine . | 41 |
| 6.4.2. RQ2: Đánh giá hệ thống giới thiệu lai | 42 |
| 6.4.3. RQ3: Phân tích hiệu năng | 42 |
| 6.4.4. RQ4: Đánh giá hiệu quả bảo vệ quyền riêng tư | 43 |
| 6.5. Thảo luận và Hạn chế | 43 |
| 6.5.1. Thảo luận | 43 |
| 6.5.2. Hạn chế | 44 |
| 6.6. Kết quả thực nghiệm chi tiết | 45 |
| 6.6.1. 1. Hiệu năng quy trình tạo tài khoản (Upsert Pipeline) | 45 |
| 6.6.2. 2. Phân tích kết quả giới thiệu và Hiệu quả tối ưu hoá | 45 |
| 6.6.3. 3. Logic cập nhật ELO thực tế | 46 |
| Kết luận và Hướng phát triển | 47 |
| 6.7. Tổng kết kết quả đạt được | 47 |
| 6.8. Hạn chế và hướng phát triển tương lai | 47 |
| Công bô liên quan | 49 |
| Tài liệu tham khảo | 50 |
| Phụ lục | 52 |
| A. Thông tin về mã nguồn và cơ sở dữ liệu | 52 |
| B. Bộ câu hỏi Big Five (Phiên bản Tiếng Anh) | 52 |
| C. Mã nguồn một số hàm quan trọng | 55 |
| C.1. Mã hóa và Giải mã dữ liệu (score-crypto) | 55 |
| C.2. Thuật toán Giới thiệu người dùng (recommend-users) | 57 |
| C.3. Cập nhật ELO và Tương tác (match-update) | 58 |

| | | |
|------|------------------------------------------------------------------------------|----|
| C.4. | Chính sách bảo mật cơ sở dữ liệu (RLS Policies) | 58 |
| D. | Mã nguồn kịch bản kiểm thử (Benchmarks) | 59 |
| D.1. | Script tạo dữ liệu mẫu và đo hiệu năng Upsert (seedMockProfiles.ts) | 60 |
| D.2. | Script kiểm thử kịch bản Viewer (benchmark_scenarios.ts) . . | 62 |
| E. | Kỹ thuật tối ưu hóa cơ sở dữ liệu | 64 |

DANH MỤC HÌNH ẢNH

| | | |
|----------|-------------------------------------------------------------------------------------------|----|
| Hình 1.1 | Bối cảnh ứng dụng mạng xã hội và nhu cầu kết nối theo tính cách . | 2 |
| Hình 1.2 | Rủi ro khi xử lý dữ liệu tính cách theo mô hình tập trung | 3 |
| Hình 1.3 | Quy trình giới thiệu Big Five sang vector PCA-4 | 5 |
| Hình 1.4 | Luồng mã hoá AES-GCM và lưu trữ dữ liệu tính cách | 6 |
| Hình 2.1 | Quy trình tổng thể của hệ thống Twins | 10 |
| Hình 2.2 | Sơ đồ trọng số tính giới thiệu xếp hạng | 13 |
| Hình 2.3 | Đoạn log tại edge function thể hiện quá trình mã hoá dữ liệu được gửi từ người dùng. | 14 |
| Hình 3.1 | Minh họa mô hình MBTI và cách phân nhóm tính cách | 16 |
| Hình 3.2 | Minh họa cấu trúc 6 yếu tố của HEXACO | 17 |
| Hình 3.3 | Minh họa tiêu chí lựa chọn PCA-4 | 19 |
| Hình 4.1 | Định dạng đầu vào/dầu ra của AES-GCM | 23 |
| Hình 4.2 | Luồng giao diện và vị trí tổng hợp điểm Big Five | 24 |
| Hình 4.3 | Ví dụ chi phí tính toán khi dùng RSA cho payload nhỏ | 26 |
| Hình 4.4 | So sánh dữ liệu băm và dữ liệu có thể giải mã | 27 |
| Hình 4.5 | Minh họa độ phức tạp của mã hoá đồng hình | 27 |
| Hình 4.6 | So sánh sự riêng tư biệt lập và mã hoá dữ liệu cá nhân | 28 |
| Hình 4.7 | Nhật ký Edge Function khi mã hoá và giải mã dữ liệu | 29 |
| Hình 4.8 | Ví dụ dữ liệu đã mã hoá của Big Five trong bảng profiles | 29 |
| Hình 4.9 | Luồng mã hoá dữ liệu sở thích và lưu trữ vector nhúng | 30 |
| Hình 5.1 | Ví dụ ELO phản ánh hành vi xã giao qua chuỗi tương tác | 32 |
| Hình 5.2 | Mức độ tương đồng ngữ nghĩa của hai từ được so sánh bằng cosine similarity. | 34 |
| Hình 5.3 | TF-IDF vượt trội ở khả năng tìm kiếm từ khoá quan trọng. | 35 |
| Hình 5.4 | Word2Vec vượt trội trong việc tìm quan hệ ngữ nghĩa giữa các từ. | 36 |

DANH MỤC BẢNG

| | |
|---------------------------------------------------------------------|----|
| Bảng 6.1 Độ trễ phản hồi trung bình của các tác vụ hệ thống | 42 |
| Bảng 6.2 Hiệu năng quy trình tạo tài khoản | 45 |
| Bảng 6.3 So sánh hiệu năng giới thiệu trước và sau tối ưu hoá | 45 |
| Bảng 6.4 Kết quả xếp hạng thực tế sau khi tối ưu hoá | 46 |

DANH MỤC GIẢI THUẬT

| | | |
|----------------|-----------------------------------------------------------|----|
| Thuật toán 2.1 | Tính toán kỳ vọng thắng trong mô hình Elo | 11 |
| Thuật toán 2.2 | Quy tắc cập nhật điểm ELO hợp tác | 12 |
| Thuật toán 2.3 | Tính toán độ gần (proximity) ELO | 12 |
| Thuật toán 2.4 | Thuật toán tính điểm giới thiệu tổng hợp | 12 |
| Thuật toán 3.1 | Phép chiếu PCA giảm chiều dữ liệu tính cách | 20 |
| Thuật toán 4.1 | Quy trình mã hoá và xác thực dữ liệu bằng AES-256-GCM . | 22 |
| Thuật toán 5.1 | Công thức tính độ tương đồng cosine (Cosine Similarity) . | 34 |
| Thuật toán 1.1 | Hiện thực hàm mã hoá và giải mã điểm tính cách | 56 |
| Thuật toán 1.2 | Hiện thực hàm giới thiệu người dùng lai | 57 |
| Thuật toán 1.3 | Hiện thực hàm cập nhật điểm ELO và tương tác | 58 |
| Thuật toán 1.4 | Script tạo dữ liệu mẫu và đo hiệu năng | 61 |
| Thuật toán 1.5 | Script kiểm thử kịch bản giới thiệu | 63 |
| Thuật toán 1.6 | Mã nguồn SQL tối ưu hoá cơ sở dữ liệu | 64 |

Chương 1

Giới thiệu

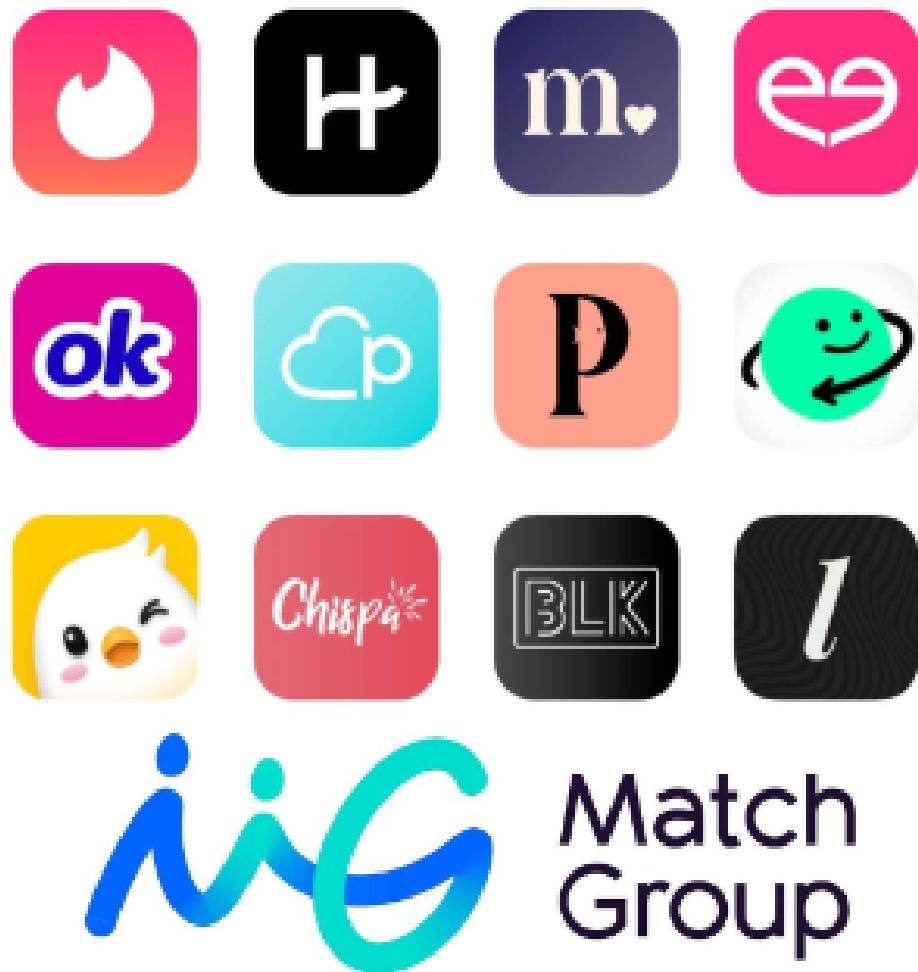
1.1. Bối cảnh và vấn đề

1.1.1. Mạng xã hội và nhu cầu kết nối theo tính cách

Twins là một ứng dụng mạng xã hội theo hướng bán khép kín, tập trung vào các cộng đồng nhỏ và chất lượng. Ứng dụng hướng tới việc tìm bạn có tính cách và sở thích tương đồng, lấy cảm hứng từ cơ chế lướt của Tinder và yếu tố kết nối thân mật của Locket. Khác với những nền tảng đại trà, Twins ưu tiên kết nối có chiều sâu thay vì số lượng tương tác. Mục tiêu này dẫn tới việc giảm bớt các tín hiệu bề mặt và tăng trọng số cho các yếu tố phản ánh đặc trưng cá nhân ổn định hơn. Về mặt trải nghiệm, người dùng được dẫn qua một chuỗi câu hỏi ngắn gọn để trích xuất tính cách, sau đó dùng kết quả này như một “dấu vân tinh cách” phục vụ giới thiệu và phân nhóm.

Các nền tảng mạng xã hội và ứng dụng kết nối hiện nay thường tối ưu cho tốc độ ghép cặp và số lượt tương tác, dựa trên yếu tố vị trí, sở thích bề mặt hoặc mạng bạn bè sẵn có. Cách tiếp cận này tạo ra nhiều kết quả, nhưng chưa chắc dẫn tới sự tương hợp lâu dài. Trong khi đó, các mô hình tính cách như Big-5 được xem là khung tham chiếu ổn định, có khả năng giải thích xu hướng hành vi và mức độ phù hợp giữa các cá nhân [1,2].

Ở góc nhìn của đề tài, nhu cầu kết nối theo tính cách có ý nghĩa vì nó gắn với các đặc trưng ít thay đổi theo thời gian, nên phù hợp cho bài toán giới thiệu dài hạn. Lựa chọn này cũng tránh việc phụ thuộc quá nhiều vào dữ liệu tương tác ngắn hạn, vốn dễ bị ảnh hưởng bởi bối cảnh, tâm trạng hoặc hiệu ứng thuật toán. Hình [Hình 1.1](#) minh họa bối cảnh ứng dụng và mục tiêu kết nối theo tính cách.



Hình 1.1 — Bối cảnh ứng dụng mạng xã hội và nhu cầu kết nối theo tính cách

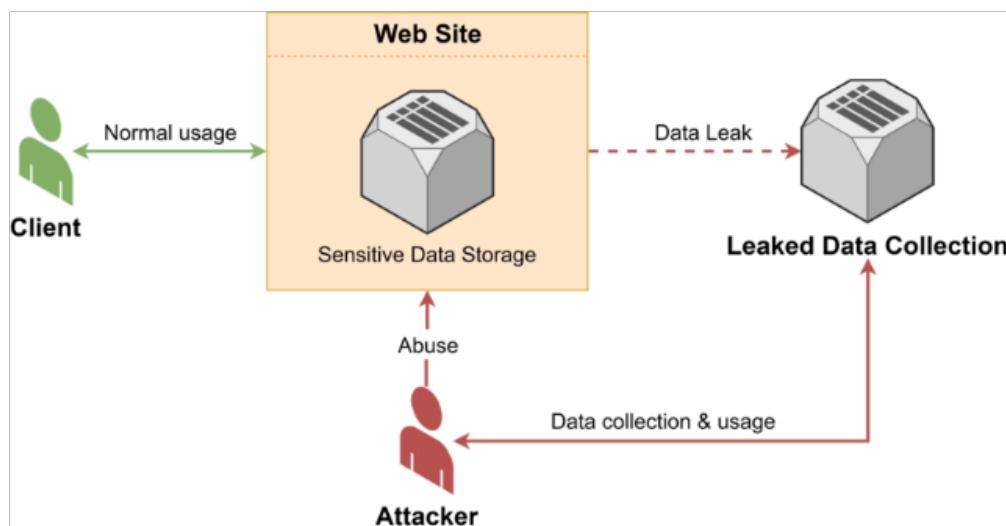
1.1.2. Rủi ro dữ liệu tính cách và yêu cầu bảo vệ

Dữ liệu tính cách có thể được suy diễn từ hành vi số hoặc từ bài trắc nghiệm, và thường được xem là dữ liệu nhạy cảm vì nó liên quan trực tiếp đến xu hướng tâm lý và hành vi của người dùng. Nhiều nghiên cứu chỉ ra rằng đặc điểm tính cách có thể dự đoán từ dữ liệu số và có mức độ ổn định cao [3]. Đồng thời, các đặc điểm này có thể bị khai thác để tác động đến hành vi, ví dụ trong các kịch bản thao túng nội dung hoặc quảng cáo cá nhân hóa quá mức [4]. Việc thu thập và lưu trữ tập trung vì thế cần được xem xét cẩn trọng về quyền riêng tư.

Trong những năm gần đây, nhiều nền tảng lớn liên tục bị cơ quan quản lý chỉ trích và xử phạt vì vi phạm quyền riêng tư. Ví dụ, FTC đã áp mức phạt 5 tỉ USD với Facebook vì các vi phạm về dữ liệu cá nhân [5]. Ở châu Âu, CNIL

áp phạt Google vì thiếu minh bạch và không có cơ sở pháp lý đầy đủ cho việc xử lý dữ liệu [6]. Các vụ việc này cho thấy áp lực pháp lý ngày càng tăng đối với những hệ thống thu thập dữ liệu người dùng quy mô lớn. Trong bối cảnh đó, việc thiết kế một quy trình (pipeline) có cơ chế bảo vệ dữ liệu ngay từ đầu là nhu cầu thực tế, không chỉ là lựa chọn kỹ thuật.

Trong bối cảnh đó, đề tài đặt ra yêu cầu bảo vệ dữ liệu tính cách ở mức tương tự như các loại dữ liệu nhạy cảm khác (tin nhắn, mật khẩu). Thay vì để dữ liệu gốc tồn tại dạng văn bản thuần (plaintext) trên máy chủ, hệ thống cần có cơ chế chuyển đổi và mã hoá để giảm thiểu rủi ro rò rỉ. Hình [Hình 1.2](#) mô tả các rủi ro chính khi xử lý dữ liệu tính cách theo mô hình tập trung.



Hình 1.2 — Rủi ro khi xử lý dữ liệu tính cách theo mô hình tập trung

1.2. Mục tiêu và phạm vi

1.2.1. Mục tiêu chính

Mục tiêu của đề tài là xây dựng một quy trình giới thiệu và bảo vệ dữ liệu tính cách, trong đó dữ liệu gốc được xử lý trên thiết bị, chuyển sang biểu diễn gọn hơn, và chỉ lưu trữ trên máy chủ dưới dạng mã hoá. Bên cạnh đó, hệ thống vẫn phải giữ khả năng so khớp và giới thiệu người dùng một cách hiệu quả.

Các mục tiêu chính gồm:

- Xây dựng cơ chế chuyển đổi điểm Big Five sang không gian đặc trưng nhỏ gọn bằng Phân tích Thành phần chính (Principal Component Analysis - PCA) với 4 chiều (PCA-4).
- Thiết kế cơ chế mã hoá theo Chuẩn mã hóa tiên tiến ở chế độ Galois/Counter (AES-GCM) để bảo vệ dữ liệu tính cách khi lưu trữ.
- Duy trì khả năng so khớp dựa trên độ tương đồng cosine (cosine similarity) để phục vụ quy trình giới thiệu.

1.2.2. Phạm vi thực hiện

Đề tài tập trung vào khía cạnh chuyển đổi dữ liệu và bảo mật, không đi sâu vào triển khai giao diện hay tối ưu hóa trải nghiệm người dùng. Phạm vi hệ thống bao gồm:

- Thiết bị người dùng thực hiện chấm điểm Big Five và chuyển đổi PCA-4.
- Một hàm thực thi biên (Edge Function) chịu trách nhiệm mã hoá và giải mã bằng AES-GCM.
- Cơ sở dữ liệu lưu trữ vector PCA và dữ liệu đã mã hoá (ciphertext) thay vì dữ liệu thô.

Ngoài ra, từ các biểu diễn đã chuyển đổi này, hệ thống giới thiệu sẽ khai thác thêm các nguồn dữ liệu đã được nhúng vector (embedding) từ sở thích và tương tác, nhằm tạo ra kết quả giới thiệu có ý nghĩa thực tế nhưng vẫn giữ được nguyên tắc bảo mật thông tin cá nhân.

1.3. Bài toán và cách tiếp cận

1.3.1. Bài toán chuyển đổi dữ liệu tính cách

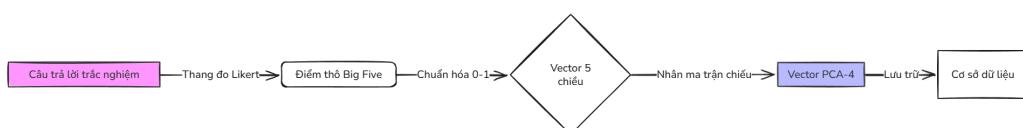
Bài toán đặt ra là chuyển đổi vector Big Five 5 chiều thành biểu diễn nhỏ gọn nhưng vẫn giữ được tính phân biệt đủ cao cho việc so khớp. Có nhiều hướng thay thế như dùng mô hình nhúng ngữ nghĩa hoặc học sâu, nhưng các hướng này thường yêu cầu dữ liệu huấn luyện lớn hơn và khó giải thích.

Trong đề tài, PCA được chọn vì Big Five là mô hình tâm lý chuẩn hóa, đã có dữ liệu công khai quy mô lớn và ổn định theo quốc gia [1,2]. PCA cho phép giảm chiều mà vẫn giữ được phần lớn phương sai. Kết quả từ notebook thực nghiệm cho thấy PCA-4 giữ khoảng hơn 90% phương sai của dữ liệu gốc, trong khi PCA-2 hoặc PCA-3 mất đáng kể thông tin [7]. Hình [Hình 1.3](#) mô tả quy trình giới thiệu Big Five sang PCA-4.

Một điểm quan trọng là tính cách khác với ngôn ngữ tự nhiên. Đối với ngôn ngữ, việc nhúng văn bản thường dựa trên các mô hình ngữ nghĩa (semantic model) lớn vì nội dung có tính mơ hồ, đa nghĩa và phụ thuộc ngữ cảnh. Trong khi đó, Big Five đã là một mô hình tâm lý chuẩn hóa, có cấu trúc dữ liệu rõ ràng và nguồn dữ liệu đủ lớn. Vì vậy PCA và độ tương đồng cosine phù hợp hơn cho phân tích cách, giúp giữ tính diễn giải và ổn định. Các mô hình ngữ nghĩa vẫn được sử dụng cho phần sở thích (hobbies), nơi dữ liệu là văn bản tự do và cần ánh xạ ngữ nghĩa.

Nói cách khác, để tài không tìm cách “học lại” tính cách bằng mô hình ngôn ngữ, mà tận dụng một hệ đo đã có sẵn trong tâm lý học. PCA chỉ là bước nén và sắp xếp lại thông tin, không thay đổi ý nghĩa gốc của Big Five. Điều này giúp tránh lệch chuẩn khi dùng mô hình học sâu khó giải thích, đồng thời giảm phụ thuộc vào dữ liệu huấn luyện nội bộ. Tính cách vì thế được xử lý như một tín hiệu có cấu trúc, còn ngôn ngữ được xử lý như tín hiệu mở.

Ở cấp độ thu thập, hệ thống sử dụng bộ câu hỏi tính cách lớn hơn, sau đó chọn ngẫu nhiên 25 câu cho mỗi lượt làm bài. Mỗi 5 câu đại diện cho một nhóm đặc điểm (trait), và điểm số được cộng hoặc trừ tùy theo hướng câu hỏi. Mô hình không phụ thuộc nội dung câu hỏi mà chỉ quan tâm đến hướng (key) và trait tương ứng. Cách tiếp cận này giúp duy trì tính nhất quán của thang đo trong khi giảm tải thời gian trả lời cho người dùng.

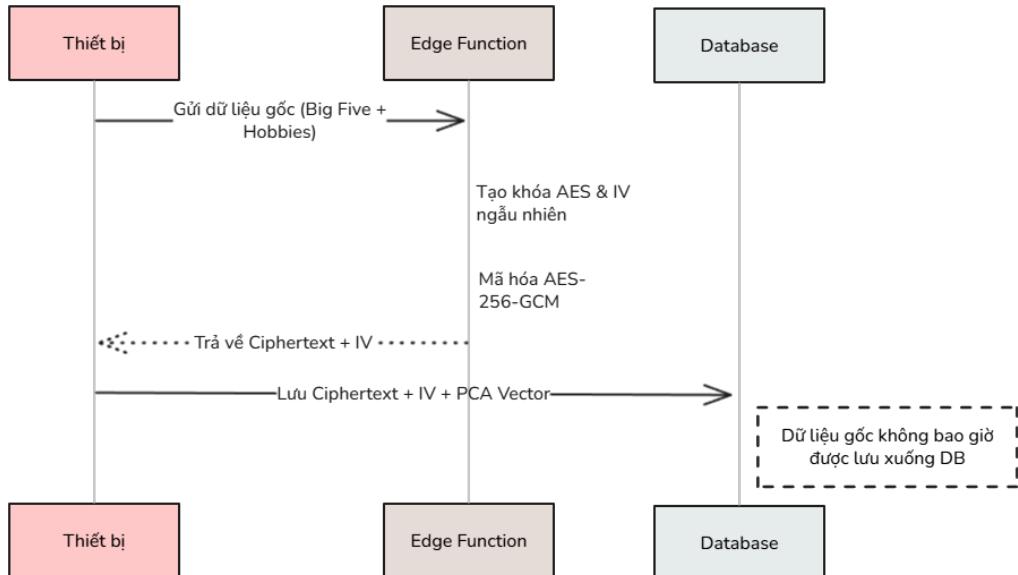


Hình 1.3 — Quy trình giới thiệu Big Five sang vector PCA-4

1.3.2. Bài toán bảo mật dữ liệu

PCA không phải cơ chế bảo mật. Các thành phần PCA có thể bị suy ngược gần đúng nếu biết tham số mô hình. Vì vậy, dữ liệu gốc vẫn cần được mã hoá. Trong số các phương án, AES-256-GCM được chọn vì phù hợp với khối lượng dữ liệu (payload) nhỏ, tốc độ cao và có tính toàn vẹn dữ liệu (integrity) nhờ GCM [8]. So với RSA hoặc Bcrypt, AES-GCM ít tốn tài nguyên hơn cho dữ liệu dạng JSON, và phù hợp với mô hình hàm thực thi bên.

Trong hệ thống, khóa AES chỉ nằm ở phía máy chủ (Edge Function). Thiết bị người dùng không giữ khóa, nhằm tránh nguy cơ bị trích xuất từ ứng dụng và vẫn cho phép khôi phục dữ liệu khi đăng nhập lại trên thiết bị khác. Hình [Hình 1.4](#) mô tả luồng mã hoá và lưu trữ dữ liệu tính cách.



Hình 1.4 — Luồng mã hoá AES-GCM và lưu trữ dữ liệu tính cách

1.4. Đóng góp chính

1.4.1. Đóng góp về mô hình chuyển đổi

Để tài xài dựng quy trình chuyển đổi Big Five sang PCA-4 chạy trên thiết bị, đảm bảo giảm kích thước dữ liệu nhưng vẫn giữ phần lớn thông tin. Hệ số PCA được huấn luyện trên tập dữ liệu công khai quy mô lớn, giúp kết quả có tính ổn định và tái lập.

1.4.2. Đóng góp về bảo mật

Để tài đề xuất cơ chế mã hoá AES-256-GCM qua Edge Function, đảm bảo dữ liệu gốc không lưu dưới dạng văn bản thuần trên cơ sở dữ liệu. Cách tiếp cận này cân bằng giữa khả năng so khớp và yêu cầu bảo mật dữ liệu nhạy cảm.

1.4.3. Đóng góp về tài liệu kỹ thuật và minh chứng

Toàn bộ mã nguồn cốt lõi của ứng dụng, bao gồm quy trình xử lý trên thiết bị, các hàm thực thi biên và cấu trúc cơ sở dữ liệu, được cung cấp đính kèm cùng

báo cáo này. Đây là nguồn tài liệu minh chứng cho quá trình hiện thực, đồng thời phục vụ công tác thẩm định và đối soát kết quả của Hội đồng.

1.5. Cấu trúc của báo cáo

Phần còn lại của báo cáo được trình bày như sau:

- **Chương 2:** Trình bày quy trình tổng thể của hệ thống Twins, từ thu thập dữ liệu đến giới thiệu.
- **Chương 3:** Phân tích chi tiết PCA-4, dữ liệu huấn luyện và cách chuyển đổi.
- (Dự kiến) Chương 4: Trình bày cơ chế bảo mật và luồng mã hoá/giải mã.
- (Dự kiến) Chương 5: Trình bày hệ giới thiệu (PCA, ELO, hobbies) và cách tính trọng số.
- (Dự kiến) Chương 6: Thực nghiệm và đánh giá hệ thống.
- (Dự kiến) Chương 7: Kết luận và hướng phát triển.

Chương 2

Tổng quan quy trình hệ thống

2.1. Mục tiêu của chương

Chương này trình bày quy trình (pipeline) tổng thể của hệ thống Twins, theo thứ tự từ thu thập dữ liệu trên thiết bị, chuyển đổi và bảo mật, đến giới thiệu người dùng. Mục tiêu là mô tả rõ các tác nhân tham gia, dữ liệu vào ra ở mỗi bước và cách các điểm số được kết hợp thành một giới thiệu xếp hạng cuối cùng. Các chương sau sẽ đi sâu vào từng thành phần. Trong đó, Chương 4 tập trung vào bảo mật và mã hoá dữ liệu, còn Chương 5 trình bày chi tiết hệ giới thiệu và các công thức xếp hạng.

2.2. Các nguồn dữ liệu đầu vào

2.2.1. Bộ câu hỏi Big Five và cách lấy mẫu

Hệ thống sử dụng tập câu hỏi Big Five lớn, được tổng hợp từ các bộ câu hỏi chuẩn như IPIP 50 và các biến thể đã được công bố rộng rãi [9]. Mỗi lượt làm bài chọn ngẫu nhiên 25 câu từ một tập hợp (pool) 150 câu, trong đó mỗi 5 câu đại diện cho một đặc điểm (trait). Mỗi câu hỏi có hướng cộng hoặc trừ vào trait tương ứng, do đó mô hình không phụ thuộc nội dung câu hỏi mà chỉ phụ thuộc vào hướng (key) và trait của câu hỏi.

Cách lấy mẫu này giúp giảm thời gian làm bài, đồng thời vẫn giữ được cấu trúc cân bằng giữa các trait. Trên thực tế, hệ thống chỉ cần biết hai thông tin cho mỗi câu: thuộc tính trait nào và hướng tính điểm (cộng hay trừ). Nội dung câu hỏi được giữ để đảm bảo ngữ cảnh người dùng, nhưng không ảnh hưởng đến mô hình chuyển đổi. Trong Chương 3 sẽ trình bày chi tiết cách tính điểm từ thang Likert và quy trình chuẩn hóa.

2.2.2. Dữ liệu khảo sát công khai cho PCA

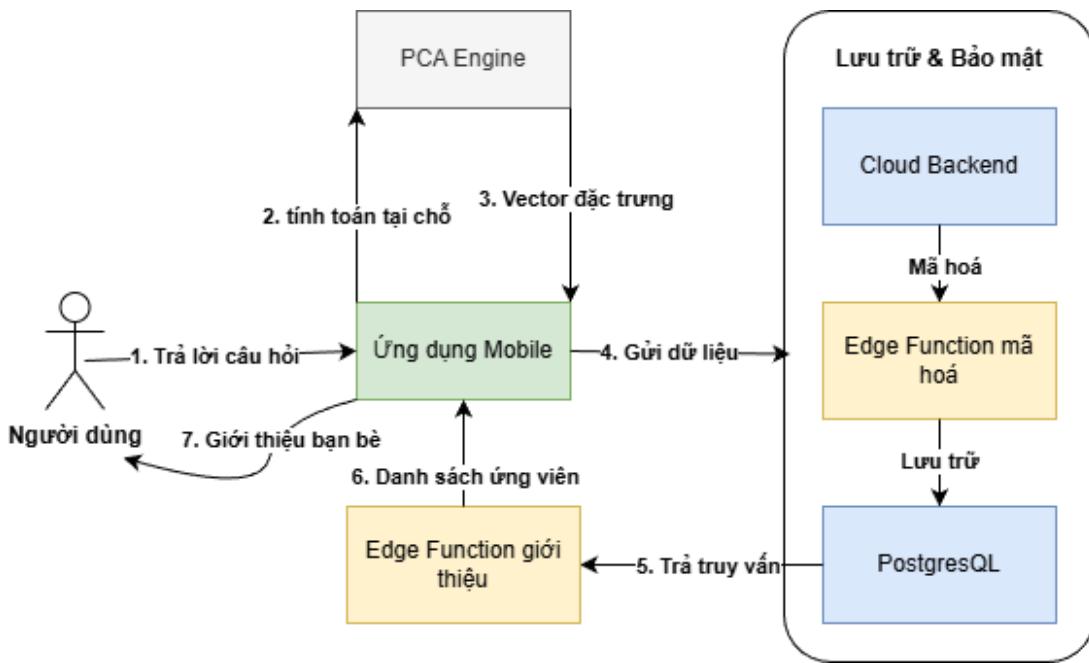
Để huấn luyện PCA, đề tài sử dụng tập dữ liệu Big Five công khai với hơn 300 nghìn mẫu từ nhiều quốc gia [7]. Dữ liệu đã được chuẩn hóa về thang 0-1 cho từng trait, phù hợp cho việc ước lượng các thành phần chính. Các kết quả giải thích phương sai sẽ được nêu ở Chương 3. Đây là lợi thế của Big Five: dữ liệu chuẩn hóa, quy mô lớn và đã được sử dụng rộng rãi trong nghiên cứu, nên PCA có thể học được cấu trúc phân bố ổn định.

2.2.3. Dữ liệu sở thích (hobbies)

Sở thích người dùng được nhập dưới dạng văn bản ngắn. Văn bản này không dùng để lưu trữ trực tiếp, mà được chuyển thành vector 384 chiều thông qua mô hình nhúng ngữ nghĩa (semantic embedding) từ Jina. Lý do dùng phương pháp nhúng là để so khớp nội dung sở thích theo ngữ nghĩa thay vì so khớp từ khóa đơn thuần. Cách làm này cho phép các sở thích có nghĩa gần nhau (ví dụ “chạy bộ” và “jogging”) vẫn được đánh giá tương đồng. Chi tiết quy trình nhúng và luồng mã hoá dữ liệu sở thích sẽ được mô tả ở Chương 5.

2.3. Tổng quan quy trình và tác nhân

Hệ thống có ba tác nhân chính: thiết bị người dùng, Edge Function và cơ sở dữ liệu. Hình [Hình 2.1](#) mô tả quy trình tổng thể từ thu thập dữ liệu đến giới thiệu.



Hình 2.1 — Quy trình tổng thể của hệ thống Twins

Các bước chính gồm:

- Thiết bị người dùng trả lời 25 câu hỏi, chấm điểm Big Five và chuẩn hóa về thang 0-1.
- Thiết bị chuyển đổi PCA-4 bằng tham số đã huấn luyện sẵn.
- Thiết bị gửi dữ liệu Big Five gốc tới Edge Function để mã hóa AES-256-GCM.
- Cơ sở dữ liệu lưu trữ pca_dim1..4 và ciphertext (b5_cipher, b5_iv).
- Dữ liệu sở thích được nhúng thành vector 384 chiều, mã hóa, và lưu trữ tương tự.
- Hệ giới thiệu lấy vector PCA, ELO và vector sở thích để tính giới thiệu xếp hạng.

2.4. Đề xuất phân mảnh địa lý trong quy trình giới thiệu

Khi số lượng người dùng tăng lớn, việc so khớp theo tổ hợp từng cặp sẽ làm chi phí tính toán tăng nhanh. Một hướng giảm tải là phân mảnh địa lý (geosharding), tức chia người dùng theo vùng địa lý hoặc cụm vị trí, sau đó ưu tiên so khớp trong cùng một phân mảnh (shard). Cách này phổ biến ở các ứng dụng hẹn hò vì nó giảm số lượng cặp cần so sánh và tăng tốc phản hồi.

Trong đề tài, geosharding được xem là bước tối ưu hóa dài hạn, chưa ưu tiên ở giai đoạn thử nghiệm. Khi lượng người dùng đủ lớn và chi phí tính toán trở thành nút thắt, hệ thống đề xuất bổ sung tầng shard theo vùng để giới hạn không gian tìm kiếm. Điều này không thay đổi công thức giới thiệu, nhưng làm giảm khói lượng tính toán cho mỗi lượt giới thiệu.

2.5. Mô hình giới thiệu và trọng số trong giới thiệu

2.5.1. Điểm tương đồng tính cách (PCA)

Vector PCA-4 được dùng để đo tương đồng giữa hai người dùng bằng cosine similarity. Phương pháp này phù hợp vì đo góc giữa hai vector, ít bị ảnh hưởng bởi độ lớn tuyệt đối và ổn định khi dữ liệu đã chuẩn hóa [10]. Công thức cosine similarity sẽ được trình bày chi tiết ở Chương 5.

2.5.2. ELO từ tương tác like/skip

Hệ thống dùng điểm ELO như một thước đo xã giao, phản ánh mức độ tương tác qua hành vi like và skip. Điểm ELO được cập nhật theo kỳ vọng thắng thua trong mô hình Elo gốc, nhưng được điều chỉnh để phù hợp với ngữ cảnh kết nối xã hội [11]. Trong hệ thống:

- Like: cả hai phía tăng nhẹ.
- Skip: chỉ người chủ động skip bị trừ.

Điểm ELO không phải thước đo hấp dẫn tuyệt đối, mà là tín hiệu phụ để gom nhóm người dùng có mức tương tác tương đồng. ELO trong Twins là hệ số ẩn, được cập nhật sau mỗi lần tương tác và bị giới hạn (clamp) trong khoảng 800 đến 2000. Lưu ý rằng cách cập nhật này tạo xu hướng lạm phát điểm ELO theo thời gian, vì lượt “like” làm cả hai phía tăng điểm. Tuy vậy, mục đích chính không phải cạnh tranh, mà là đảm bảo người dùng có mức xã giao gần nhau được ưu tiên gấp nhau hơn.

Trong công thức gốc, kỳ vọng thắng được tính bởi:

Thuật toán 2.1 — Tính toán kỳ vọng thắng trong mô hình Elo

$$E_a = \frac{1}{1 + 10^{\frac{R_b - R_a}{400}}}$$

Sau đó cập nhật theo $R_{a'} = R_a + K(S_a - E_a)$. Trong Twins, kết quả like được coi là một tín hiệu hợp tác nên cả hai phía tăng nhẹ, còn skip chỉ trừ phía chủ động. Cụ thể, với $K=12$ và được giới hạn trong [800, 2000]:

Thuật toán 2.2 — Quy tắc cập nhật điểm ELO hợp tác

- Like: $R_{a'} = \text{clamp}(R_a + K(1 - E_a)), R_{b'} = \text{clamp}(R_b + K(1 - E_b))$
- Skip: $R_{a'} = \text{clamp}(R_a + K(0 - E_a)), R_{b'} = R_b$

Bên cạnh đó, hệ giới thiệu sử dụng hệ số gần nhau ELO để ưu tiên mức xâ giao tương đồng:

Thuật toán 2.3 — Tính toán độ gần (proximity) ELO

$$p = \exp\left(-|\Delta R| \frac{1}{\sigma}\right)$$

trong đó $\sigma = 400$.

2.5.3. Embedding sở thích và cosine similarity

Sở thích người dùng được chuyển thành vector 384 chiều thông qua mô hình nhúng ngữ nghĩa. Cosine similarity được dùng để đo độ gần về sở thích, thay vì so khớp từ khóa. Cách làm này cho phép hai người dùng dùng từ khác nhau nhưng có ý nghĩa gần nhau vẫn được đánh giá cao hơn.

2.5.4. Trọng số tổng hợp

Giới thiệu xếp hạng cuối cùng được tính theo trọng số của PCA, ELO và hobbies dựa trên cấu hình hệ thống:

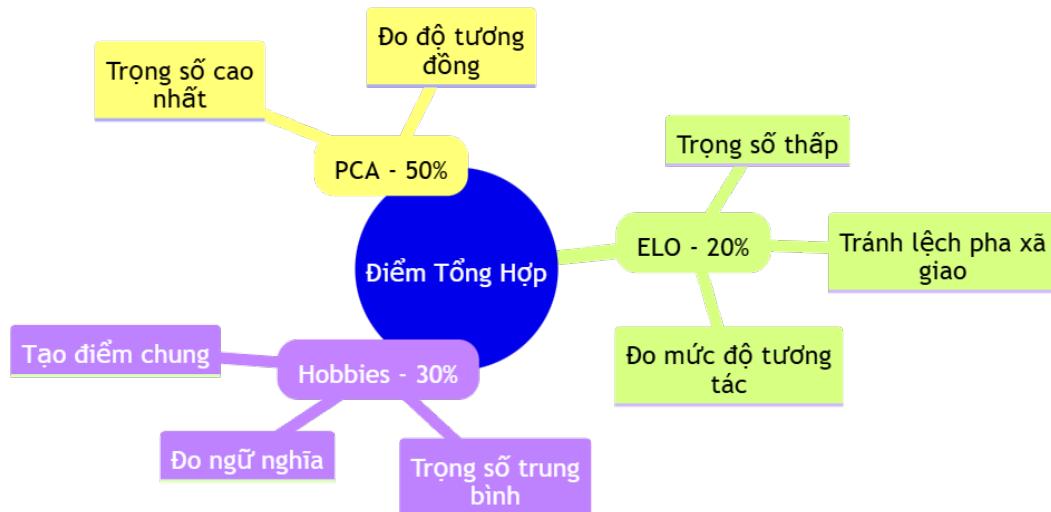
Thuật toán 2.4 — Thuật toán tính điểm giới thiệu tổng hợp

- **Trường hợp không sử dụng sở thích:**
 - ▶ Nếu ELO bật: $S = 0.8 \cdot P + 0.2 \cdot p$
 - ▶ Nếu ELO tắt: $S = P$
- **Trường hợp sử dụng sở thích:**
 - ▶ Nếu ELO bật: $S = 0.5 \cdot P + 0.2 \cdot p + 0.3 \cdot H$
 - ▶ Nếu ELO tắt: $S = 0.55 \cdot P + 0.45 \cdot H$

Trong đó: P là độ tương đồng PCA, p là hệ số gần nhau ELO, H là độ tương đồng sở thích.

Để minh họa, xét ba người dùng A, B, C khi A đang tìm giới thiệu. Giả sử A có PCA tương đồng với B và C gần bằng nhau (ví dụ 0.90), nhưng B có sở thích

gần hơn (hobbies 0.85) trong khi C có ELO gần hơn (proximity 1.0 so với 0.7). Trong cấu hình có ELO và hobbies, giới thiệu cuối có thể làm B đứng trước nếu lợi thế sở thích lớn hơn lợi thế ELO. Trường hợp ngược lại, nếu B và C ngang nhau về hobbies, thì C sẽ được ưu tiên do proximity cao hơn.



Hình 2.2 — Sơ đồ trọng số tính giới thiệu xếp hạng

2.6. Luồng dữ liệu chi tiết theo tác nhân

2.6.1. Thiết bị người dùng

Thiết bị thực hiện các bước sau:

- Thu thập câu trả lời và chấm điểm Big Five.
- Chuẩn hóa và chuyển đổi PCA-4.
- Gửi dữ liệu thô tới Edge Function để mã hoá.
- Gửi văn bản sở thích để tạo vector, rồi lưu ciphertext và vector nhúng.

2.6.2. Edge Function

Edge Function đảm nhận:

- Mã hoá/giải mã Big Five bằng AES-256-GCM.
- Gọi dịch vụ nhúng để sinh vector sở thích.
- Trả về ciphertext, iv và vector nhúng cho thiết bị.

2.6.3. Cơ sở dữ liệu

Cơ sở dữ liệu lưu trữ:

- Vector PCA (pca_dim1..4).
- Ciphertext và iv cho Big Five (b5_cipher, b5_iv).
- Ciphertext cho hobbies và vector nhúng.

Edge function Log

```

29 Dec 25 03:16:30 ⚡ INFO [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] encrypt success (2ms)
29 Dec 25 03:16:30 ⚡ INFO [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] mode=encrypt hint=generic_array
29 Dec 25 03:16:30 ⚡ INFO [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] encrypting: ["Anime", "Gardening", "Hiking"]...
29 Dec 25 03:16:30 ⚡ INFO [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] incoming POST
29 Dec 25 03:16:30 ⚡ INFO Listening on http://localhost:9999/

```

Hình 2.3 — Đoạn log tại edge function thể hiện quá trình mã hoá dữ liệu được gửi từ người dùng.

2.7. Cấu trúc các chương tiếp theo

Để đi sâu vào từng thành phần của quy trình, các chương tiếp theo của báo cáo được cấu trúc như sau, tách biệt rõ ràng các phần hiện thực và thực nghiệm để tăng tính mạch lạc:

- **Chương 3: Chuyển đổi dữ liệu tính cách (PCA-4)**, tập trung vào phần lõi của việc xử lý và biểu diễn dữ liệu tính cách.
- **Chương 4: Bảo mật và mã hoá dữ liệu**, trình bày chi tiết kiến trúc bảo mật, một thành phần quan trọng của hệ thống.
- **Chương 5: Hệ giới thiệu và cơ chế xếp hạng**, mô tả logic nghiệp vụ của việc kết hợp các tín hiệu để đưa ra giới thiệu cuối cùng.
- **Chương 6: Thực nghiệm và Đánh giá**, dành riêng cho việc kiểm chứng và đánh giá toàn bộ hệ thống dựa trên các câu hỏi nghiên cứu đã đề ra.

Cách phân chia này giúp người đọc theo dõi chi tiết từng khía cạnh của việc “Hiện thực” (Chương 3, 4, 5) trước khi đi vào phần “Thực nghiệm” (Chương 6).

Chương 3

Chuyển đổi dữ liệu tính cách (PCA-4)

3.1. Mục tiêu của chương

Chương này trình bày chi tiết quy trình chuyển đổi dữ liệu Big Five sang vector PCA-4, bao gồm cách chuẩn hóa điểm, cách huấn luyện PCA và cách triển khai trên thiết bị. Mục đích là làm rõ vì sao PCA-4 được chọn thay vì PCA-2/3 hoặc các mô hình nhúng vector khác.

3.2. Big Five trong bối cảnh các mô hình tính cách

Trong tâm lý học có nhiều khung mô tả tính cách, không có mô hình nào tuyệt đối hoàn hảo. Big Five được sử dụng vì đã có lịch sử nghiên cứu dài, hệ thống câu hỏi chuẩn hóa và dữ liệu công khai phong phú. So với các mô hình khác như MBTI hoặc HEXACO, Big Five có ưu thế về tính tái lập và độ phủ dữ liệu, phù hợp cho bài toán chuyển đổi số liệu quy mô lớn [2,12]. Do đó, để tài chấp nhận giới hạn của mô hình nhưng coi Big Five là lựa chọn thực tế nhất để làm nền cho quy trình chuyển đổi dữ liệu.

3.2.1. Mô hình Chỉ báo Phân loại Myers-Briggs (MBTI)

MBTI phân loại người dùng theo các cặp đối lập, tạo ra 16 nhóm tính cách. Cách biểu diễn này dễ truyền thông nhưng thiên về phân loại rời rạc, trong khi dữ liệu thực tế thường có phân bố liên tục. Với bài toán giới thiệu cần đo mức độ gần nhau, dạng nhãn rời rạc làm giảm khả năng xếp hạng chi tiết và khó phản ánh mức độ “gần” giữa hai cá nhân. MBTI cũng có vấn đề về độ ổn định theo thời gian, nhiều người thay đổi nhóm khi làm lại bài test. Điều này làm cho dữ liệu khó tái lập và khó dùng cho quy trình so khớp dài hạn. Ngoài ra, MBTI ít có dữ liệu mở quy mô lớn theo chuẩn hóa số điểm, nên khó dùng cho chuyển đổi

PCA và huấn luyện ổn định. Ví dụ, hai người thuộc nhóm INFP và ENFP có thể khác nhau mạnh về hướng ngoại nhưng vẫn bị xem là hai nhãn rìa rạc. Hình [Hình 3.1](#) minh họa cách MBTI chia nhóm tính cách.

| | | | |
|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| ESTJ Tj Ambition Sj Discipline Se Experience Te Pragmatism | ESTP Sp Spontaneity Tp Inventiveness Se Experience Te Pragmatism | ESFP Sp Spontaneity Fp Honesty Se Experience Fe Romantic | ESFJ Sj Discipline Fj Kindness Se Experience Fe Romantic |
| ISTJ Si History Ti Accuracy Sj Discipline Tj Ambition | ISTP Si History Ti Accuracy Sp Spontaneity Tp Inventiveness | ISFP Si History Fi Harmony Sp Spontaneity Fp Honesty | ISFJ Si History Fi Harmony Sj Discipline Fj Kindness |
| INTJ Ni Philosophy Ti Accuracy Nj Vision Tj Ambition | INTP Ni Philosophy Ti Accuracy Np Variation Tp Inventiveness | INFP Ni Philosophy Fi Harmony Np Variation Fp Honesty | INFJ Ni Philosophy Fi Harmony Nj Vision Fj Kindness |
| ENTJ Ne Opportunity Te Pragmatism Nj Vision Tj Ambition | ENTP Ne Opportunity Te Pragmatism Np Variation Tp Inventiveness | ENFP Ne Opportunity Fe Romance Np Variation Fp Honesty | ENFJ Ne Opportunity Fe Romance Nj Vision Fj Kindness |

Hình 3.1 — Minh họa mô hình MBTI và cách phân nhóm tính cách

3.2.2. Mô hình tính cách HEXACO

HEXACO mở rộng Big Five bằng cách thêm yếu tố Trung thực-Khiêm tốn (Honesty-Humility). Mô hình này có giá trị về mặt học thuật, nhưng dữ liệu mở và bộ câu hỏi chuẩn hóa không phổ biến bằng Big Five. Việc thêm một đặc điểm (trait) thứ sáu làm tăng số câu hỏi cần thiết để giữ cân bằng độ tin cậy. Điều này gây áp lực lên trải nghiệm người dùng di động, vì thời gian trả lời dài hơn. Ngoài ra, chuyển đổi từ HEXACO sang dạng PCA sẽ cần dữ liệu huấn luyện riêng, trong khi dữ liệu chuẩn không nhiều bằng Big Five. Ví dụ, nếu chỉ dùng 25 câu, mỗi trait sẽ bị giảm số câu đánh giá, làm tăng nhiễu đo lường. Do đó

HEXACO được xem là lựa chọn tham khảo hơn là lựa chọn chính cho đề tài. Hình [Hình 3.2](#) minh họa cấu trúc HEXACO.



Hình 3.2 — Minh họa cấu trúc 6 yếu tố của HEXACO

3.3. Chuẩn hóa điểm Big Five

3.3.1. Thang đo và hướng câu hỏi

Mỗi câu trả lời được chấm theo thang Likert 1–5. Với câu hỏi hướng dương, điểm giữ nguyên thứ tự 1→5. Với câu hỏi hướng âm, điểm được đảo chiều. Sau đó các điểm trong cùng một trait được cộng lại và chuẩn hóa về thang 0–1. Cách chuẩn hóa này giúp các trait có cùng thang đo, phù hợp cho PCA và so khớp cosine.

3.3.2. Ví dụ định dạng dữ liệu đầu vào

Sau bước chuẩn hóa, mỗi người dùng có một vector 5 chiều theo thứ tự trait cố định:

```
x = [Extraversion, Agreeableness, Conscientiousness, Emotional Stability, Intellect]
```

Ví dụ một người dùng có thể có:

```
x = [0.68, 0.55, 0.72, 0.60, 0.47]
```

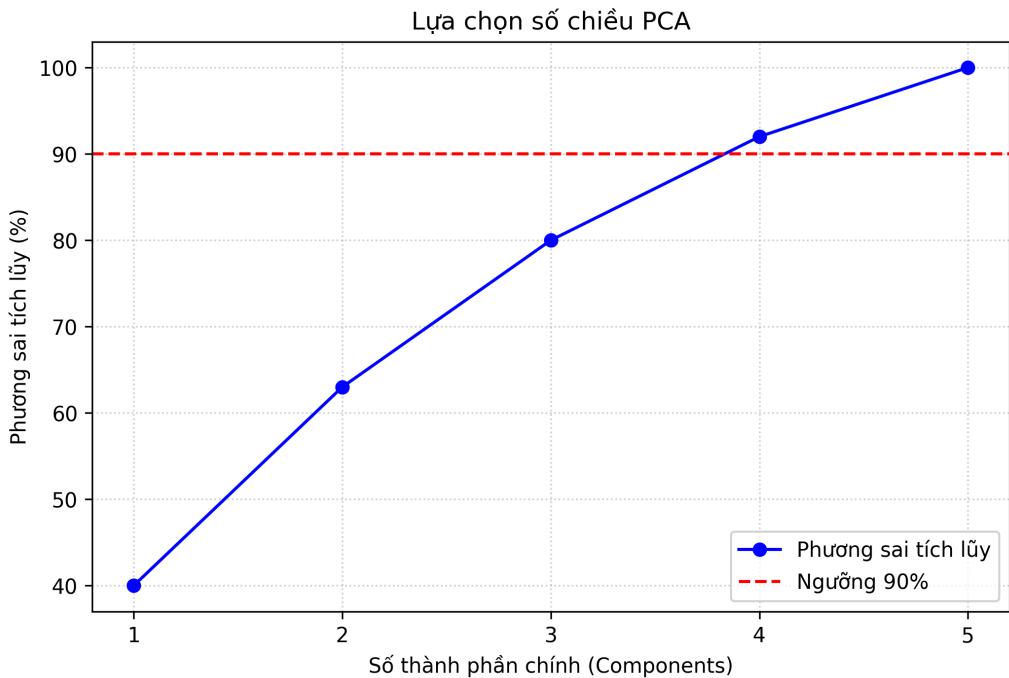
Đây là dạng dữ liệu đầu vào cho bước PCA.

3.3.3. Vì sao chọn PCA-4 sau khi chuẩn hóa

Chuẩn hóa đưa dữ liệu Big Five về cùng thang đo, giúp mỗi trait đóng góp cân bằng khi so khớp và khi học PCA. Tuy vậy, chuẩn hóa không giải quyết vấn đề dư thừa thông tin giữa các trait. PCA được dùng để rút gọn chiều và tách các trục phương sai lớn nhất. Trong khi PCA-2 hoặc PCA-3 làm mất đáng kể thông tin, PCA-4 là điểm cân bằng tối ưu: giảm chiều từ 5 xuống 4 nhưng vẫn giữ phần lớn phương sai, giúp hệ giới thiệu hoạt động ổn định khi đo độ tương đồng cosine.

3.4. Đề xuất PCA-4

Đề tài đề xuất PCA-4 như mức giảm chiều tối ưu cho Big Five trong bối cảnh giới thiệu bạn bè. Giảm từ 5 xuống 4 chiều giúp tiết kiệm lưu trữ mà vẫn giữ phần lớn cấu trúc dữ liệu. PCA-4 cũng là dạng biểu diễn dễ triển khai trên thiết bị với phép nhân ma trận thuận. Mức giảm nhẹ này giúp hạn chế rủi ro mất thông tin so với PCA-2 hoặc PCA-3. Ngoài ra, PCA-4 giữ được tính diễn giải tương đối, phù hợp với việc so sánh độ tương đồng cosine ổn định. Hình [Hình 3.3](#) trình bày một minh họa quyết định chọn PCA-4 dựa trên phương sai.



Hình 3.3 — Minh họa tiêu chí lựa chọn PCA-4

Trong notebook thực nghiệm, PCA-2 chỉ giữ khoảng 63% phương sai, PCA-3 khoảng 80%, trong khi PCA-4 giữ hơn 90% phương sai dữ liệu gốc. Sự chênh lệch này ảnh hưởng trực tiếp đến khả năng phân biệt giữa các người dùng khi so khớp. Vì vậy PCA-4 được chọn để giảm mất thông tin mà vẫn đảm bảo kích thước nhỏ gọn.

3.5. Huấn luyện PCA

3.5.1. Nguồn dữ liệu và quy mô

PCA được huấn luyện từ tập dữ liệu Big Five công khai quy mô lớn, sử dụng tệp `big_five_scores.csv` (khoảng 307 nghìn bản ghi) [7,13]. Dữ liệu bao gồm thông tin theo quốc gia và đã chuẩn hóa điểm về thang 0–1. Trong quá trình thăm dò, thống kê cho thấy dữ liệu trải rộng khoảng hơn 190 quốc gia và vùng lãnh thổ, với các phân phối điểm khá ổn định giữa các nhóm quốc gia lớn. Một số bản ghi thiếu nhận quốc gia, nhưng các cột điểm số vẫn đầy đủ, vì vậy không ảnh hưởng đến việc huấn luyện PCA.

Phân tích Dữ liệu Khám phá (Exploratory Data Analysis - EDA) trong notebook cho thấy chênh lệch trung bình giữa các quốc gia tồn tại nhưng không đủ lớn để cần một mô hình riêng theo vùng. Do đó, PCA được huấn luyện trên

toàn bộ tập dữ liệu để nắm bắt phương sai tổng thể. Đây là quyết định thực tế giúp mô hình ổn định và tái lập, đồng thời tránh việc phải duy trì nhiều mô hình theo vùng.

Để tài không huấn luyện mô hình học sâu cho tính cách vì mục tiêu chính là biến đổi và nén dữ liệu đã có cấu trúc. PCA cho phép giữ tính giải thích, dễ triển khai trên thiết bị và không cần dữ liệu nhãn bổ sung. Nếu dùng mô hình phức tạp hơn, chi phí huấn luyện và suy diễn sẽ tăng, trong khi lợi ích bổ sung không rõ ràng vì dữ liệu đã được chuẩn hóa.

3.5.2. Công thức chiểu PCA

PCA thực hiện phép chiểu tuyến tính trên dữ liệu đã được trừ đi giá trị trung bình. Với vector đầu vào x (dài 5), ta có:

$$z = (x - \mu) \times W^T \quad (3.1)$$

trong đó μ là vector trung bình (mean) và W là ma trận chứa các thành phần chính (components) [14]. Vector z là PCA-4 và được lưu dưới dạng 4 chiều. Giải thuật [Thuật toán 3.1](#) mô tả phép chiểu và định dạng đầu ra.

Thuật toán 3.1 — Phép chiểu PCA giảm chiểu dữ liệu tính cách

$$z = (x - \mu) \times W = (x_1 \ x_2 \ x_3 \ x_4 \ x_5) \times \begin{pmatrix} w_{1,1} & w_{1,2} & w_{1,3} & w_{1,4} \\ w_{2,1} & w_{2,2} & w_{2,3} & w_{2,4} \\ \vdots & \vdots & \vdots & \vdots \\ w_{5,1} & w_{5,2} & w_{5,3} & w_{5,4} \end{pmatrix} = (z_1 \ z_2 \ z_3 \ z_4)$$

3.6. Triển khai PCA trên thiết bị

3.6.1. Cách triển khai

Thay vì chạy mô hình học sâu, PCA-4 được triển khai bằng phép nhân ma trận thuần trên thiết bị. Các hệ số trung bình và thành phần chính được trích từ notebook huấn luyện và lưu cố định trong ứng dụng. Cách này giảm phụ thuộc vào các thư viện học máy (Machine Learning - ML) và hạn chế kích thước gói ứng dụng (bundle).

3.6.2. Định dạng lưu trữ

Kết quả PCA-4 được lưu dưới dạng 4 trường số: pca_dim1..pca_dim4. Các giá trị này được lưu song song với ciphertext của Big Five. Việc lưu PCA dạng số thực giúp tính độ tương đồng cosine trực tiếp ở phía máy chủ khi giới thiệu.

3.7. Thảo luận lựa chọn PCA

PCA là phép biến đổi tuyến tính, có thể giải thích và kiểm soát. Các lựa chọn thay thế như nhúng vector học sâu hoặc nhúng ngữ nghĩa (semantic embedding) không phù hợp vì dữ liệu tính cách đã có cấu trúc rõ ràng và ít phụ thuộc ngôn ngữ. Ngoài ra, PCA giúp duy trì tính ổn định giữa các phiên bản, tránh lệch kết quả do thay đổi mô hình.

Chương 4

Bảo mật và mã hoá dữ liệu

4.1. Mục tiêu của chương

Chương này trình bày cách dữ liệu được nhập từ góc độ người dùng, cách dữ liệu được chuyển đổi và mã hoá trước khi lưu trữ, cùng với lý do lựa chọn cơ chế AES-256-GCM. Trọng tâm là luồng dữ liệu và các tác nhân, không đi sâu vào mã nguồn chi tiết.

4.2. Tổng quan về cơ chế AES-GCM

4.2.1. Nguyên lý cơ bản

AES là thuật toán mã hoá đối xứng khối, hoạt động trên các khối dữ liệu cố định và cần một khóa

chung cho cả quá trình mã hoá lẫn giải mã. Chế độ GCM (Galois/Counter Mode) kết hợp

giữa mã hoá dạng bộ đếm (counter mode) và cơ chế xác thực dữ liệu.

Thuật toán 4.1 — Quy trình mã hoá và xác thực dữ liệu bằng AES-256-GCM

1. **Khởi tạo:** Tạo khóa bí mật K và vector khởi tạo ngẫu nhiên IV .
2. **Mã hoá:** $C = E_{K(IV,P)}$, trong đó P là văn bản thuần, C là văn bản mã hoá.
3. **Xác thực:** Sinh thẻ xác thực T dựa trên K, IV và C .
4. **Lưu trữ:** Lưu cặp (C, IV, T) vào cơ sở dữ liệu.
5. **Giải mã:** Kiểm tra T trước khi khôi phục $P = D_{K(IV,C)}$.

Nhờ đó, ngoài dữ liệu đã mã hoá (ciphertext), hệ thống còn có thể kiểm tra tính toàn vẹn (integrity) của dữ liệu [8].

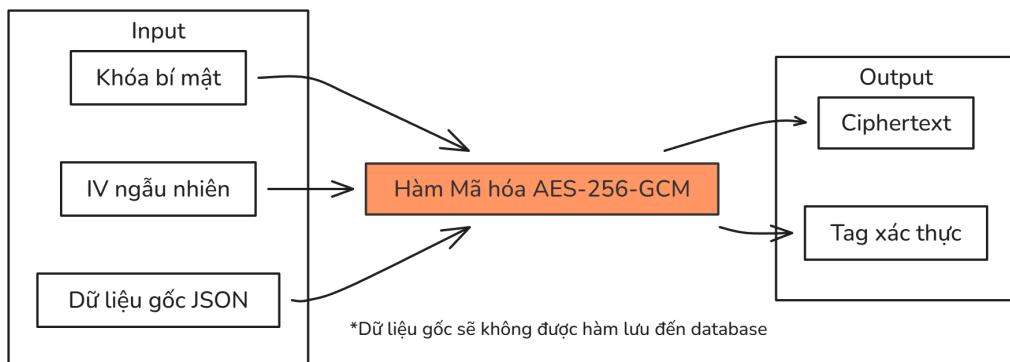
Trong ngữ cảnh dữ liệu tính cách, yếu tố này rất quan trọng để đảm bảo dữ liệu không bị thay đổi trái phép mà không bị phát hiện.

Một phiên làm việc AES-GCM tạo ra thêm thẻ xác thực (authentication tag), giúp phát hiện bất kỳ sự thay đổi nào đối với dữ liệu hoặc vector khởi tạo (Initialization Vector - IV). Nếu thẻ xác thực không khớp, dữ liệu sẽ bị từ chối giải mã. Cơ chế này làm giảm nguy cơ người dùng nhận phải dữ liệu sai lệch hoặc đã bị chỉnh sửa khi truyền qua mạng. Với dữ liệu nhạy cảm như tính cách và sở thích, việc đảm bảo tính toàn vẹn quan trọng không kém việc giữ bí mật. Vì vậy, AES-GCM phù hợp hơn các chế độ chỉ mã hoá mà không đi kèm xác thực.

4.2.2. Đầu vào và đầu ra của AES-GCM

Đầu vào bao gồm dữ liệu gốc (dưới dạng JSON chứa điểm Big Five hoặc danh sách sở thích), khóa bí mật, và một IV ngẫu nhiên. Đầu ra bao gồm dữ liệu đã mã hoá (ciphertext) và IV tương ứng. Trong triển khai của đề tài, IV được lưu trữ riêng trong cơ sở dữ liệu để phục vụ quá trình giải mã sau này. Hình [Hình 4.1](#) mô tả cấu trúc đầu vào và đầu ra của quy trình này.

Việc lưu trữ thẻ xác thực đi kèm ciphertext cho phép hệ thống kiểm tra tính toàn vẹn ngay tại thời điểm giải mã. Nếu phát hiện sai lệch, hệ thống sẽ từ chối giải mã và ghi nhận lỗi, ngăn chặn việc trả về dữ liệu sai. Cách lưu trữ này bảo vệ dữ liệu cá nhân khỏi các thay đổi ngầm ở cấp độ cơ sở dữ liệu hoặc trong quá trình truyền tải.



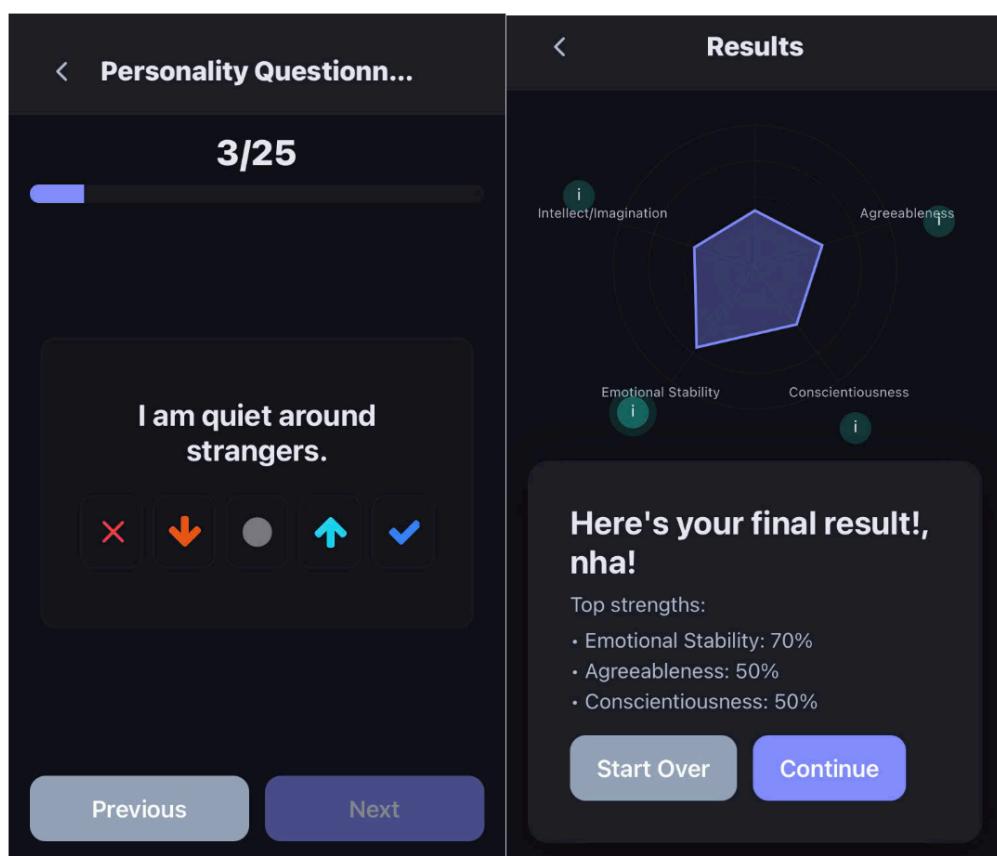
Hình 4.1 — Định dạng đầu vào/đầu ra của AES-GCM

4.3. Dữ liệu đầu vào từ góc nhìn người dùng

4.3.1. Trải nghiệm nhập liệu và ranh giới dữ liệu nhạy cảm

Người dùng thực hiện bộ câu hỏi tính cách gồm 25 câu hỏi trong một lượt. Các câu trả lời này được xem là dữ liệu nhạy cảm vì có thể dùng để suy diễn đặc trưng tâm lý. Ngay khi người dùng hoàn tất, hệ thống chỉ lưu lại các điểm số đã được tổng hợp theo mô hình Big Five, không lưu trữ câu trả lời gốc cho từng câu hỏi. Việc này giúp giảm thiểu rủi ro rò rỉ dữ liệu thô và hạn chế khả năng định danh gián tiếp.

Hình [Hình 4.2](#) minh họa bố trí giao diện và vị trí bước tổng hợp điểm trong luồng ứng dụng.



Hình 4.2 — Luồng giao diện và vị trí tổng hợp điểm Big Five

4.3.2. Chuyển đổi trên thiết bị

Sau khi tổng hợp, điểm Big Five được chuẩn hóa và chuyển đổi sang không gian PCA-4 ngay trên thiết bị người dùng. Kết quả PCA là dữ liệu đã giảm

chiều, đủ cho mục đích so khớp nhưng không thay thế hoàn toàn được dữ liệu thô. Tuy nhiên, vì PCA là phép biến đổi tuyến tính, thông tin gốc vẫn có thể bị suy ngược gần đúng nếu biết tham số mô hình. Do đó, dữ liệu gốc vẫn cần được mã hoá trước khi lưu trữ.

4.4. Mã hóa dữ liệu bằng AES-256-GCM

4.4.1. Đề xuất AES-GCM

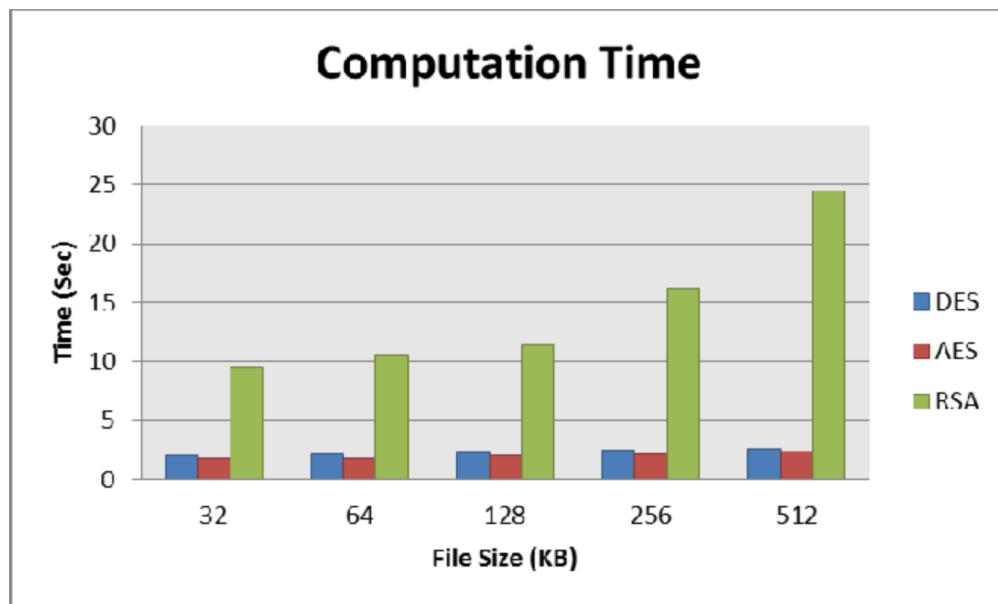
Đề tài đề xuất sử dụng AES-256-GCM làm cơ chế mã hoá chính cho dữ liệu tính cách và sở thích. Lý do là dữ liệu có kích thước nhỏ, yêu cầu tốc độ xử lý nhanh và cần khả năng giải mã để hiển thị lại trên giao diện người dùng. AES-GCM đáp ứng tốt ba yêu cầu: tốc độ, xác thực và dễ dàng triển khai trên các hàm thực thi biên (Edge Function). Cơ chế này cũng cho phép lưu trữ IV riêng biệt để tái tạo dữ liệu khi người dùng đăng nhập lại. Trong phạm vi khóa luận, AES-GCM là lựa chọn tối ưu để cân bằng giữa bảo mật và khả năng vận hành thực tế.

4.4.2. Lý do chọn AES-GCM

AES-GCM được lựa chọn vì phù hợp với các gói dữ liệu (payload) nhỏ, tốc độ cao, và tích hợp sẵn cơ chế xác thực dữ liệu (integrity) cùng lúc với mã hoá [8]. So với RSA hoặc Bcrypt, AES-GCM tiêu tốn ít nguyên hơn khi mã hoá các chuỗi JSON ngắn, và dễ dàng tích hợp trong môi trường Edge Function.

4.4.3. Lựa chọn thay thế: RSA

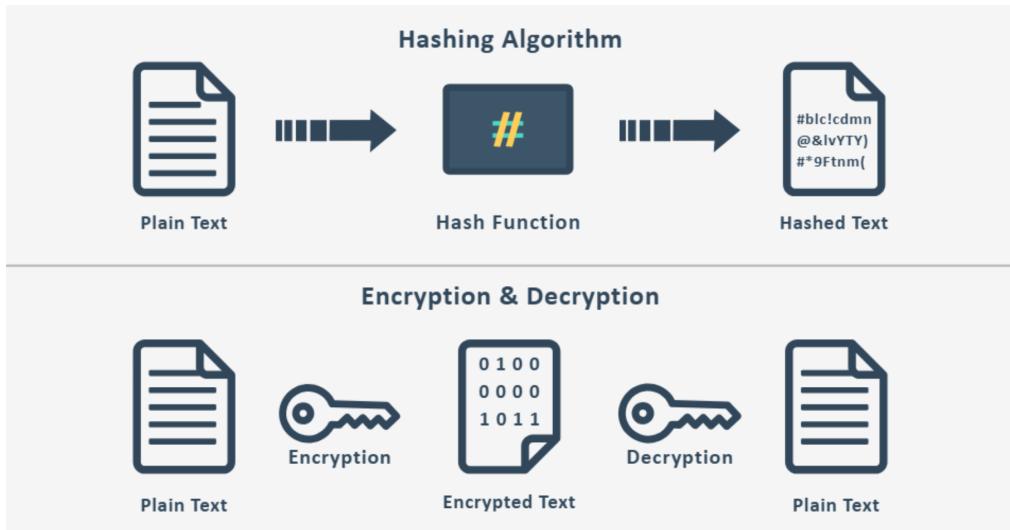
RSA là thuật toán mã hoá bất đối xứng, thường dùng để trao đổi khóa hoặc ký số [15]. Trong bối cảnh dữ liệu tính cách, RSA không phù hợp để mã hoá trực tiếp dữ liệu vì chi phí tính toán lớn và giới hạn về kích thước dữ liệu đầu vào. Nếu sử dụng RSA cho mỗi lần cập nhật hồ sơ, hệ thống sẽ gặp vấn đề về độ trễ và khó mở rộng trên thiết bị di động. Ngoài ra, RSA thường đi kèm các cơ chế đệm (padding) phức tạp, dễ phát sinh lỗi nếu không được triển khai cẩn trọng. Vì vậy, RSA được xem là phương án thay thế nhưng không phù hợp làm cơ chế mã hoá chính cho dữ liệu người dùng. Ví dụ, việc mã hoá một gói tin JSON nhỏ bằng RSA đòi hỏi nhiều bước xử lý đệm và tách khỏi, gây chậm trễ đáng kể khi người dùng cập nhật hồ sơ liên tục.



Hình 4.3 — Ví dụ chi phí tính toán khi dùng RSA cho payload nhỏ

4.4.4. Lựa chọn thay thế: Bcrypt/Scrypt

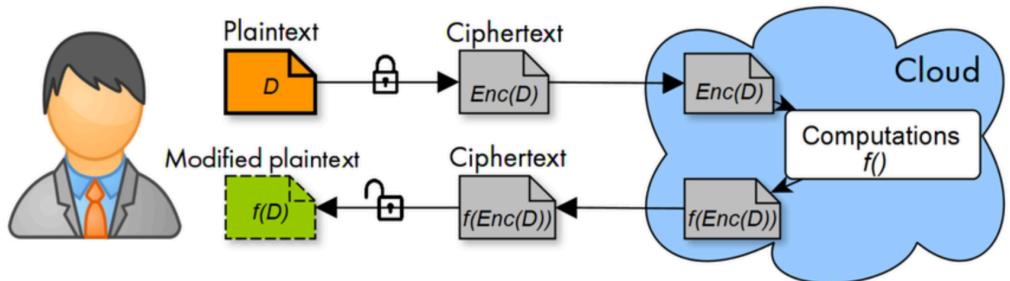
Bcrypt và Scrypt là các hàm băm mật khẩu (password hashing function) [16]. Ưu điểm của chúng là làm chậm các cuộc tấn công dò khóa (brute-force), nhưng nhược điểm là dữ liệu sau khi băm không thể giải mã để lấy lại nội dung gốc. Trong hệ thống Twins, người dùng cần xem lại kết quả tính cách và sở thích của mình, do đó yêu cầu bắt buộc là phải giải mã được dữ liệu. Nếu dùng bcrypt, hệ thống chỉ có thể so khớp chuỗi băm mà không thể trả lại dữ liệu gốc cho giao diện. Điều này đi ngược lại yêu cầu về trải nghiệm người dùng và giới hạn chức năng của ứng dụng. Vì vậy, các hàm băm này không phù hợp. Ví dụ, sở thích “chạy bộ” sau khi băm sẽ trở thành một chuỗi ký tự ngẫu nhiên và không thể khôi phục để hiển thị lại là “chạy bộ”.



Hình 4.4 — So sánh dữ liệu băm và dữ liệu có thể giải mã

4.4.5. Lựa chọn thay thế: Homomorphic encryption

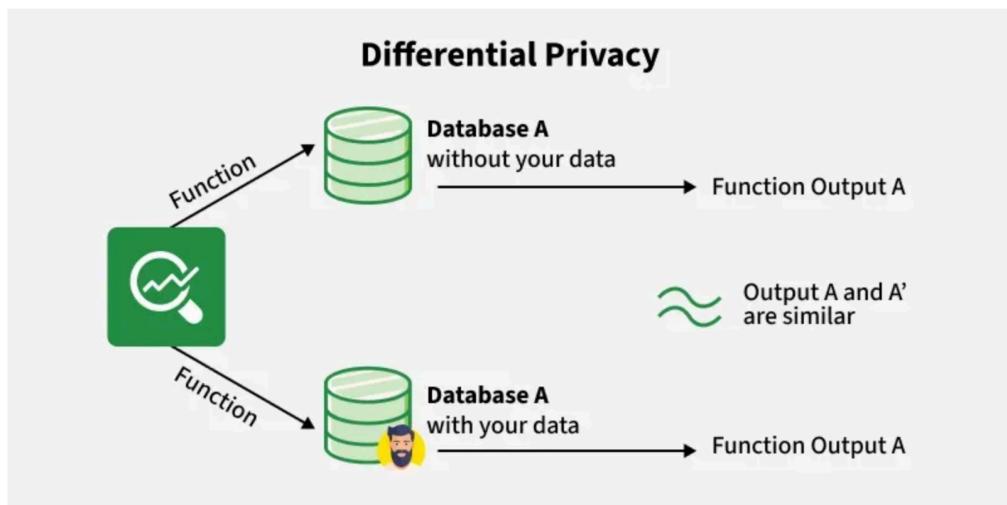
Mã hoá đồng hình (Homomorphic encryption) cho phép thực hiện tính toán trực tiếp trên dữ liệu đã mã hoá mà không cần giải mã [17]. Đây là hướng đi rất mạnh về bảo mật, nhưng chi phí tính toán cực kỳ cao và việc triển khai rất phức tạp. Với bài toán giới thiệu cần phản hồi nhanh, việc áp dụng mã hoá đồng hình sẽ làm tăng độ trễ hệ thống và đòi hỏi hạ tầng phần cứng đặc biệt. Ngoài ra, mô hình này chưa thực sự cần thiết vì để tài không yêu cầu tính toán phức tạp trực tiếp trên dữ liệu mã hoá mà chỉ cần lưu trữ an toàn và giải mã khi cần thiết. Do đó, mã hoá đồng hình vượt quá phạm vi thực tế của khóa luận. Ví dụ, một phép so khớp cosine trên dữ liệu mã hoá đồng hình có thể chậm hơn nhiều lần so với trên dữ liệu văn bản thuần, gây trải nghiệm kém mượt mà trên thiết bị di động.



Hình 4.5 — Minh họa độ phức tạp của mã hoá đồng hình

4.4.6. Lựa chọn thay thế: Differential privacy

Sự riêng tư biệt lập (Differential privacy) tập trung vào việc ẩn danh hóa khi công bố các số liệu thống kê [18]. Phương pháp này phù hợp cho dữ liệu tổng hợp, nhưng không giải quyết được bài toán lưu trữ và giải mã dữ liệu cho từng cá nhân cụ thể. Nếu chỉ áp dụng sự riêng tư biệt lập, người dùng vẫn cần truy cập vào dữ liệu gốc của chính mình, dẫn tới vấn đề bảo mật vẫn tồn tại ở cấp độ lưu trữ. Trong hệ thống Twins, yêu cầu là bảo vệ dữ liệu của từng người nhưng vẫn cho phép họ xem lại nội dung đó. Vì vậy, sự riêng tư biệt lập được coi như một kỹ thuật bổ trợ chứ không thay thế cho AES-GCM. Ví dụ, nếu cộng thêm nhiều vào điểm Big Five để bảo vệ tính ẩn danh trong thống kê, kết quả giới thiệu cá nhân hóa cho người dùng sẽ bị giảm độ chính xác và khó giải thích.



Hình 4.6 — So sánh sự riêng tư biệt lập và mã hoá dữ liệu cá nhân

4.4.7. Vai trò của Edge Function và khóa bí mật

Khóa AES chỉ tồn tại ở phía Edge Function (máy chủ biên). Thiết bị người dùng không lưu trữ khóa này, nhằm tránh nguy cơ bị trích xuất từ ứng dụng. Đồng thời, cách thiết kế này cho phép người dùng phục hồi dữ liệu khi đăng nhập lại trên một thiết bị khác. Đây là sự cân bằng hợp lý giữa bảo mật và khả năng khôi phục dữ liệu.

Hình [Hình 4.7](#) minh họa nhật ký (log) của Edge Function cho quá trình mã hoá và giải mã.

Edge function Log

```

29 Dec 25 03:16:30 [INFO] [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] encrypt success (2ms)
29 Dec 25 03:16:30 [INFO] [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] mode=encrypt hint=generic_array
29 Dec 25 03:16:30 [INFO] [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] encrypting: ["Anime", "Gardening", "Hiking"]...
29 Dec 25 03:16:30 [INFO] [score-crypto] [be447921-1e11-416c-a9a3-5413374d2c9a] incoming POST
29 Dec 25 03:16:30 [INFO] Listening on http://localhost:9999/

```

Hình 4.7 — Nhật ký Edge Function khi mã hoá và giải mã dữ liệu

4.4.8. Lưu trữ và giới hạn truy cập

Cơ sở dữ liệu chỉ lưu trữ dữ liệu đã mã hoá và IV cho Big Five (các trường `b5_cipher`, `b5_iv`). Điều này có nghĩa là quản trị viên cơ sở dữ liệu không thể đọc trực tiếp dữ liệu tính cách dưới dạng văn bản thuần. Dữ liệu chỉ được giải mã khi người dùng đã xác thực thành công và gửi yêu cầu thông qua Edge Function. Cách làm này hạn chế nguy cơ giám sát hàng loạt (mass surveillance) từ bảng dữ liệu chưa mã hoá, đồng thời vẫn đảm bảo tính năng xem lại kết quả cho người dùng.

Hình [Hình 4.8](#) minh họa mẫu dữ liệu đã mã hoá được lưu trong cơ sở dữ liệu.

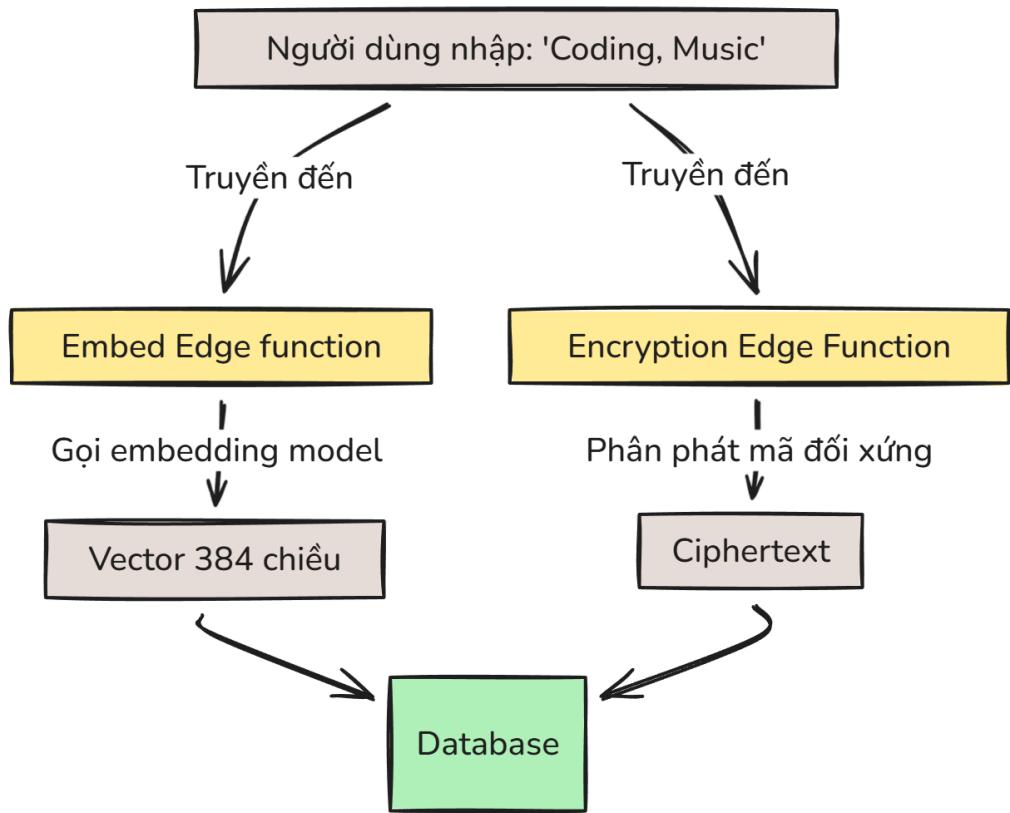
| Database store | |
|-------------------------|-------------------------------------------------------------|
| <code>hobbies_iv</code> | <code>hobbies_cipher</code> |
| HrC7WzHY3EgYx+hz | rT9Ht74YLUFk0BLi0fmGyLsdgnqTJ2f3XT22KnnR8rVPSFJivnsey5NWBM0 |
| d1CH9P4hnMJL09UJ | 6sQLDvn7RhWdEfAx2MR2j7dqhsmVivBl/6r400KKcDc/Y5cwtwHviu4= |

Hình 4.8 — Ví dụ dữ liệu đã mã hoá của Big Five trong bảng profiles

4.5. Dữ liệu sở thích và mã hóa

Sở thích người dùng được nhập dưới dạng văn bản tự do, sau đó được nhúng thành vector 384 chiều. Dữ liệu văn bản này cũng được mã hoá theo cơ chế AES-GCM tương tự như Big Five. Do đó, giao diện ứng dụng có thể hiển thị lại sở thích sau khi giải mã, nhưng cơ sở dữ liệu hoàn toàn không lưu trữ văn bản thuần.

Hình [Hình 4.9](#) mô tả luồng dữ liệu sở thích từ nhập liệu đến lưu trữ.



Hình 4.9 — Luồng mã hoá dữ liệu sở thích và lưu trữ vector nhúng

Chương 5

Hệ giới thiệu và cơ chế xếp hạng

5.1. Mục tiêu của chương

Chương này mô tả cách hệ giới thiệu kết hợp ba nguồn tín hiệu: tính cách (PCA), hành vi xã giao (ELO) và sở thích được nhúng vector (embedding hobbies). Đồng thời, chương giải thích vì sao từng tín hiệu vẫn cần thiết, ngay cả khi người dùng đã khai báo sở thích, và đi sâu vào các luận điểm thiết kế đằng sau mỗi thành phần.

5.2. Vì sao vẫn cần tính cách khi đã có sở thích

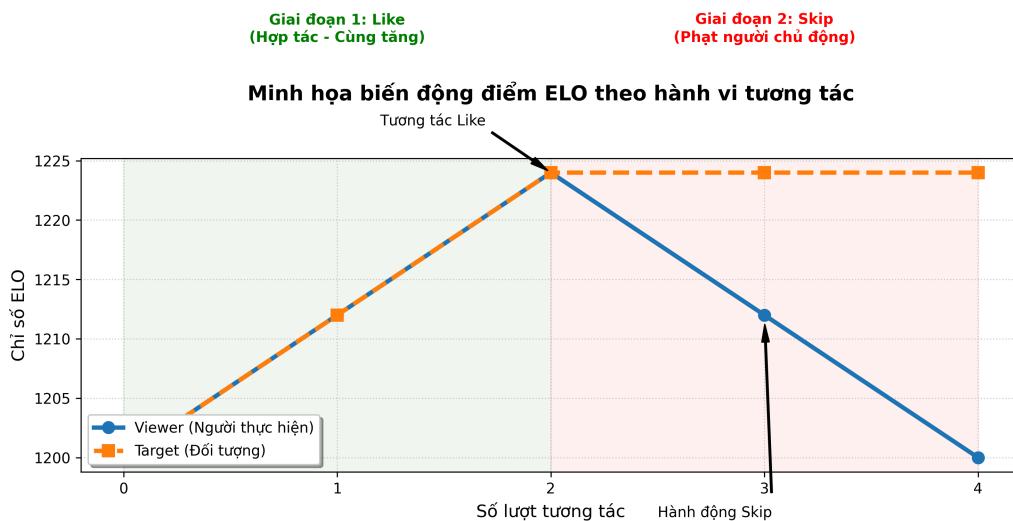
Sở thích (interests) phản ánh các chủ đề người dùng quan tâm, nhưng không đủ để mô tả mức độ tương hợp về cách suy nghĩ và hành vi. Hai người cùng thích “chụp ảnh” có thể khác nhau rõ rệt về cách giao tiếp, nhịp sống và mức độ ổn định cảm xúc. Với các kết nối dài hạn, các khác biệt này thường quan trọng hơn sở thích bề mặt. Hơn nữa, sở thích có thể mang tính thời điểm hoặc thay đổi theo xu hướng, trong khi các đặc điểm tính cách cốt lõi theo mô hình Big Five có xu hướng ổn định hơn nhiều trong suốt cuộc đời của một người trưởng thành.

Vì vậy, tính cách được xác định là **trục ổn định** (stable axis) của hệ giới thiệu, đảm bảo các kết nối có nền tảng vững chắc và chiều sâu. Sở thích đóng vai trò **trục ngữ cảnh** (contextual axis), giúp bổ trợ, phá vỡ các trường hợp hòa điểm và tìm ra các điểm chung tức thời. Việc kết hợp cả hai giúp hệ thống vừa ổn định trong dài hạn, vừa linh hoạt trong ngắn hạn.

5.3. Đề xuất thuật toán ELO

Trong hệ thống, ELO được dùng như một tín hiệu hành vi ẩn. ELO không nói người dùng “tốt” hơn hay “xấu” hơn, mà phản ánh mức độ xã giao thể hiện qua lượt like/skip. Công thức cập nhật dựa trên kỳ vọng thắng thua gốc của Elo [11], được điều chỉnh để phù hợp với bối cảnh kết nối, nơi lượt like là một tín hiệu hợp tác. Cách cập nhật chi tiết đã được mô tả ở Giải thuật [Thuật toán 2.1](#) và Giải thuật [Thuật toán 2.3](#). Việc giới hạn điểm trong khoảng 800–2000 giúp tránh việc điểm bị trôi quá xa và làm giảm tác dụng phân nhóm hành vi.

Hình [Hình 5.1](#) minh họa trực quan cách ELO phản ánh hành vi xã giao qua các chuỗi like/skip khác nhau.



Hình 5.1 — Ví dụ ELO phản ánh hành vi xã giao qua chuỗi tương tác

5.3.1. Vai trò của ELO trong hành vi xã giao

Điểm ELO phản ánh mức độ like/skip trong thực tế. Đây là tín hiệu hành vi, không phải kết quả tự khai báo. Nó đóng vai trò là một cơ chế hiệu chỉnh, giúp giảm sai lệch giữa những gì người dùng **nói** họ là (qua bài trắc nghiệm) và những gì họ **làm** (qua hành vi lượt). Khi người dùng thường xuyên skip, điểm ELO giảm và hệ thống ưu tiên giới thiệu những người có mức xã giao tương đồng.

ELO trong hệ thống là hệ số ẩn, được cập nhật sau mỗi tương tác và giới hạn trong khoảng 800–2000. Mặc dù cập nhật theo kiểu hợp tác dẫn tới lạm phát điểm, mục tiêu chính là gom nhóm hành vi thay vì xếp hạng cạnh tranh.

5.3.2. Bàn luận về thiết kế ELO

Việc điều chỉnh thuật toán ELO cho bối cảnh mạng xã hội thay vì một trò chơi đối kháng tổng bằng không (zero-sum game) là một quyết định thiết kế quan trọng.

- **Quy tắc cập nhật “hợp tác”:** Trong cờ vua, một người thắng thì người kia thua. Trong một tương tác “like”, cả hai đều có thể nhận được giá trị. Việc tăng điểm cho cả hai bên khuyến khích tương tác tích cực và tránh “trùng phạt” người được yêu thích. Ngược lại, chỉ người chủ động “skip” bị trừ điểm, vì đây là hành động đơn phương thể hiện sự không phù hợp từ phía họ.
- **Hệ số K (K-factor):** Hệ số K=12 được chọn là một giá trị tương đối nhỏ. Điều này làm cho điểm ELO thay đổi từ từ, phản ánh một quá trình xây dựng “danh tiếng xã giao” dài hạn thay vì biến động mạnh sau vài tương tác. Nó giúp điểm số ổn định hơn và tránh bị lạm dụng.
- **Cơ chế Giới hạn (Clamping) (800-2000):** Việc giới hạn điểm số trong một khoảng nhất định ngăn chặn hiện tượng “lạm phát ELO” vô hạn và giữ cho sự khác biệt về điểm số luôn nằm trong một phạm vi có ý nghĩa, đảm bảo thành phần ELO proximity trong công thức tổng hợp không trở nên quá lớn hoặc quá nhỏ.

5.4. Người sử dụng sở thích

Sở thích chỉ được dùng khi người dùng nhập đủ số lượng tối thiểu (3 mục). Điều này tránh việc dùng dữ liệu quá ít dẫn tới nhiễu hoặc thiên lệch do một sở thích đơn lẻ. Khi đủ người, vector nhúng (embedding vector) được tạo và dùng độ tương đồng cosine để tính điểm gần nhau về sở thích. Quy tắc người này cũng giúp người dùng mới không bị bất lợi nếu chưa kịp khai báo đầy đủ sở thích, tạo ra một sân chơi công bằng hơn.

5.5. Đề xuất mô hình ngữ nghĩa (semantic model)

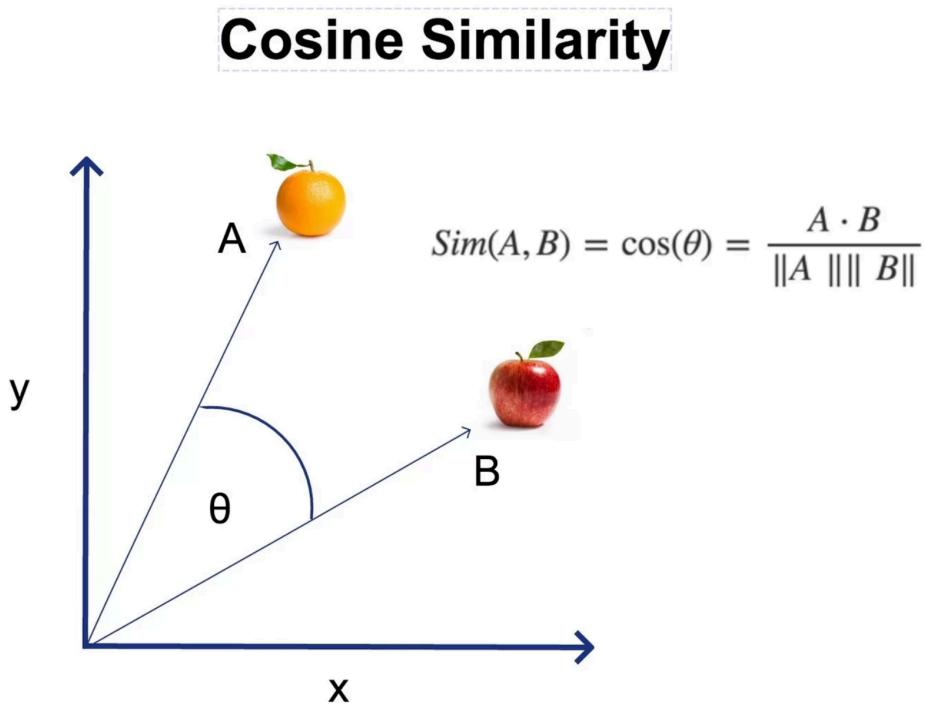
Đề tài sử dụng mô hình ngữ nghĩa của Jina (semantic model) để chuyển đổi văn bản sở thích thành vector 384 chiều. Lý do chính là khả năng nắm bắt tương đồng ngữ nghĩa thay vì trùng từ khóa, phù hợp với cách người dùng mô tả sở thích bằng nhiều cách khác nhau.

Thuật toán 5.1 — Công thức tính độ tương đồng cosine (Cosine Similarity)

$$\text{sim}(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}}$$

Mô hình kiểu nhúng câu (sentence embedding) cũng ổn định khi so khớp độ tương đồng cosine, dễ triển khai và ít tốn tài nguyên hơn so với các mô hình sinh lớn [19]. Hình [Hình 5.2](#) mô tả luồng chuyển đổi từ văn bản sang vector và cách dùng độ tương đồng cosine trong giới thiệu.

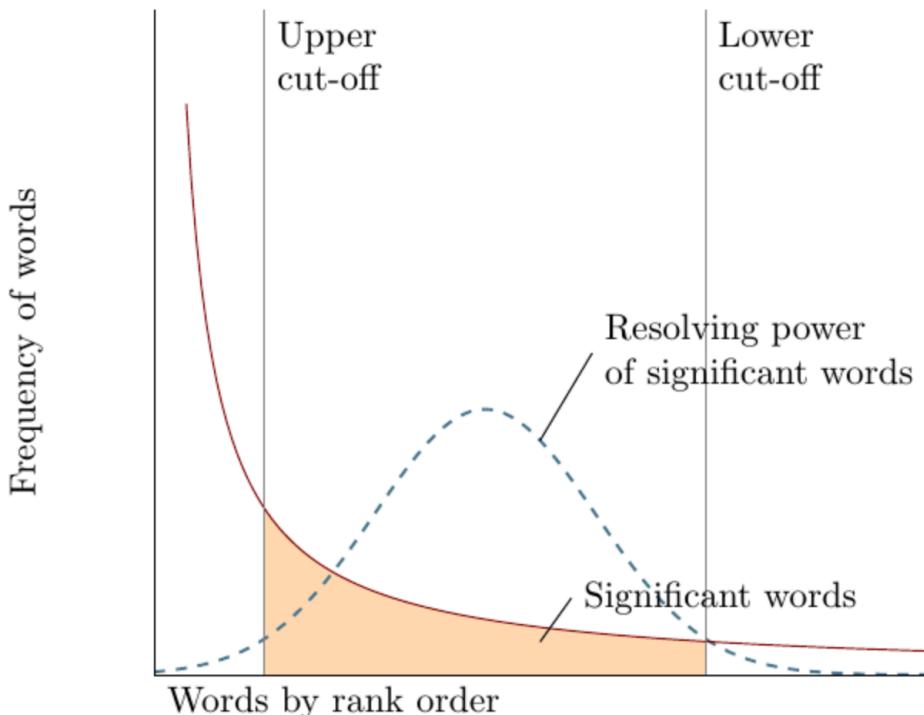
Trong triển khai hiện tại, hệ thống ghép sở thích thành một chuỗi ngắn theo mẫu `interests: ...` rồi sinh một vector duy nhất. Cách làm này là một sự đánh đổi có chủ đích giữa độ chính xác và hiệu năng. Việc chỉ có một vector cho mỗi người dùng giúp giảm chi phí so sánh xuống $O(N)$, thay vì $O(N*k^2)$ nếu mỗi người có k sở thích và phải so sánh chéo. Điều này giúp hệ thống có khả năng mở rộng tốt hơn. Quan điểm của đề tài là ưu tiên tính ổn định và khả năng mở rộng, và chỉ xem xét mô hình đa vector khi có hạ tầng đủ mạnh.



Hình 5.2 — Mức độ tương đồng ngữ nghĩa của hai từ được so sánh bằng cosine similarity.

5.5.1. Lựa chọn thay thế: TF-IDF

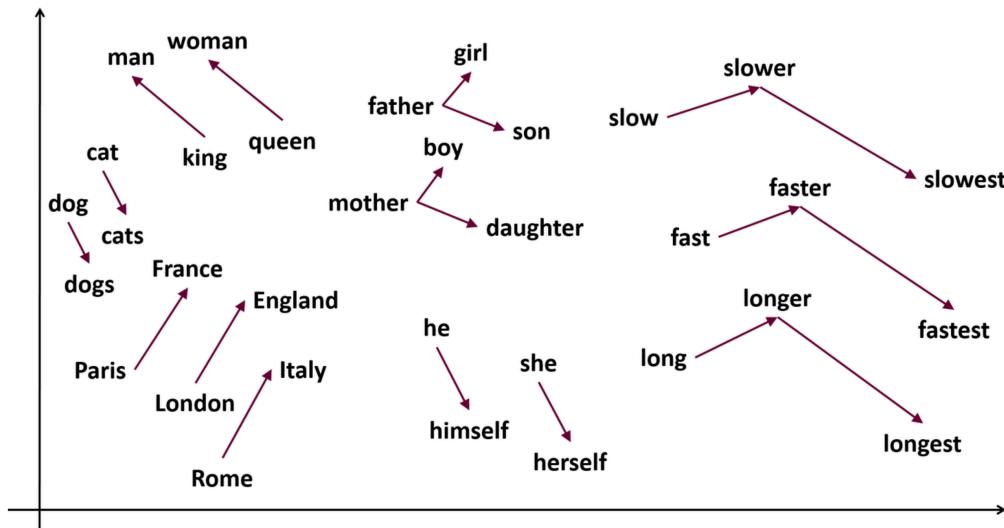
TF-IDF là cách biểu diễn văn bản theo trọng số từ khóa [10]. Điểm mạnh của TF-IDF là đơn giản, dễ giải thích, và chạy nhanh trên thiết bị. Tuy nhiên, TF-IDF không hiểu ngữ nghĩa nên khó nhận biết các từ đồng nghĩa như “jogging” và “chạy bộ”. Ngoài ra, TF-IDF tạo vector thừa và kích thước lớn, làm tăng chi phí lưu trữ và so khớp khi số lượng từ vựng tăng. Trong bối cảnh sở thích ngắn và đa dạng, TF-IDF dễ bị nhiễu bởi các từ hiếm. Vì vậy, TF-IDF được coi là lựa chọn thay thế tham khảo chứ không phù hợp làm lõi giới thiệu.



Hình 5.3 — TF-IDF vượt trội ở khả năng tìm kiếm từ khoá quan trọng.

5.5.2. Lựa chọn thay thế: Word2Vec

Word2Vec tạo vector cho từng từ dựa trên ngữ cảnh [20]. Cách này nắm bắt được một phần quan hệ ngữ nghĩa, nhưng vẫn gặp khó khi chuyển sang mức câu hoặc cụm sở thích ngắn. Người dùng thường nhập cụm như “đi phượt cuối tuần” hoặc “nấu ăn healthy”, trong khi Word2Vec cần thêm bước gộp nhiều vector (ví dụ: lấy trung bình) để đại diện cho cả cụm. Việc gộp thủ công làm mất sắc thái và không ổn định giữa các mẫu khác nhau. Do đó, các mô hình nhúng câu được ưu tiên vì xử lý trực tiếp cụm sở thích, ổn định hơn trong so khớp.



Hình 5.4 — Word2Vec vượt trội trong việc tìm quan hệ ngữ nghĩa giữa các từ.

5.6. Công thức xếp hạng tổng hợp

Hệ thống tính điểm theo các trọng số đã nêu ở Chương 2. Về bản chất, PCA là trực chính, ELO là trực hành vi, và hobbies là trực ngữ nghĩa.

Việc đặt PCA làm trực chính giúp kết quả ổn định hơn theo thời gian, vì tính cách thay đổi chậm và ít bị ảnh hưởng bởi các biến động ngắn hạn. ELO chỉ đóng vai trò điều chỉnh, tránh trường hợp hai người có tính cách gần nhau nhưng hành vi xã giao quá khác biệt. Hobbies được dùng như một tín hiệu làm mượt, giúp hệ giới thiệu nhận ra các chủ đề tương đồng mà tính cách không nắm bắt được. Cấu trúc này giảm rủi ro hệ thống chỉ dựa vào một nguồn dữ liệu duy nhất, vốn dễ gây thiên lệch hoặc thiếu đa dạng.

5.6.1. Bàn luận về trọng số

Việc lựa chọn và phân bổ các trọng số trong công thức giới thiệu tổng hợp (ví dụ: 60% PCA, 15% ELO, 25% Hobbies) là một quy trình cân nhắc kỹ lưỡng nhằm đạt được sự cân bằng giữa tính ổn định dài hạn và các yếu tố ngữ cảnh tức thời. **Xác định tính cách là nền tảng cốt lõi** đóng vai trò quyết định trong việc duy trì một mối quan hệ bền vững, do đó trọng số cho sự tương đồng PCA luôn được thiết lập ở mức ưu tiên cao nhất, thường chiếm trên 50% tổng điểm giới thiệu. Việc đặt trọng số này ở mức chủ đạo giúp hệ thống lọc ra những người dùng có “sóng não” và xu hướng tâm lý tương hợp dựa trên mô hình Big-5, từ

đó giảm thiểu rủi ro của các kết nối bề mặt vốn dễ dẫn đến sự ngắt quãng sau một thời gian ngắn tương tác.

Sử dụng hành vi xã giao làm yếu tố hiệu chỉnh thông qua điểm số ELO nhằm tinh chỉnh danh sách giới thiệu sao cho phù hợp với mức độ năng động của từng cá nhân trong thực tế. Trọng số ELO được giữ ở mức thấp nhất trong bộ quy tắc vì nó không phản ánh sự tương hợp về bản chất con người mà chỉ đóng vai trò như một bộ lọc hành vi. Cơ chế này giúp tránh tình trạng giới thiệu “lệch pha” giữa một người dùng quá năng nổ với một người dùng có xu hướng khắt khe hoặc thụ động hơn, từ đó làm mượt trải nghiệm lướt và tăng tỷ lệ phản hồi tích cực dựa trên sự tương đồng về phong cách giao tiếp.

Coi sở thích cá nhân là chất xúc tác và cầu nối ngữ cảnh để tạo ra những chủ đề trò chuyện cụ thể ngay từ giai đoạn đầu của việc kết nối. Với trọng số đáng kể nhưng thấp hơn tính cách, thành phần nhúng ngữ nghĩa của sở thích giúp phá vỡ thế hòa điểm giữa các ứng viên có độ tương đồng PCA ngang nhau, đồng thời cung cấp những giới thiệu mang tính thời điểm và thực tế cao hơn. Điều này cho phép người dùng dễ dàng tìm thấy tiếng nói chung thông qua các hoạt động hoặc đam mê cụ thể, từ đó tạo tiền đề cho việc khám phá sâu hơn về tính cách trong tương lai. Các trọng số này có thể được hiệu chỉnh linh hoạt thông qua các thử nghiệm thực tế hoặc cá nhân hóa cho từng nhóm người dùng, nhưng cấu hình hiện tại được xem là một điểm khởi đầu cân bằng, đảm bảo tính khoa học và hiệu quả của hệ thống giới thiệu.

5.6.2. Ví dụ minh họa xếp hạng

Xét người dùng A đang xem giới thiệu, với ba ứng viên B và C. Giả sử:

- PCA similarity: A-B = 0.90, A-C = 0.90 (hòa nhau).
- Hobbies similarity: A-B = 0.85, A-C = 0.55.
- ELO proximity: A-B = 0.70, A-C = 1.00.

Trong cấu hình bật cả ELO và hobbies, giới thiệu cuối của B sẽ tăng nhờ hobbies, còn C tăng nhờ ELO. Nếu trọng số hobbies lớn hơn phần chênh lệch ELO, B sẽ đứng trước. Nếu ngược lại, C sẽ đứng trước. Ví dụ này cho thấy các nguồn tín hiệu có thể phá vỡ thế hòa PCA theo các hướng khác nhau.

5.7. Bảo vệ dữ liệu sở thích và quyền riêng tư

Mặc dù UI có thể hiển thị sở thích đã giải mã, cơ sở dữ liệu không lưu văn bản thuần (plaintext). Điều này tránh việc quản trị viên có thể quét hàng loạt sở thích từ bảng dữ liệu. Người dùng chỉ thấy sở thích khi đã được xác thực và giải mã thông qua Edge Function.

Ngoài ra, việc lưu ciphertext giúp giảm rủi ro lộ dữ liệu ở cấp độ hệ quản trị. Người dùng vẫn nhìn thấy sở thích trên UI vì dữ liệu được giải mã theo phiên đăng nhập hợp lệ, nhưng cơ sở dữ liệu không có điểm tập trung văn bản thuần để khai thác hàng loạt. Đây là điểm khác biệt quan trọng so với cách lưu trữ sở thích truyền thống trong nhiều ứng dụng mạng xã hội.

Chương 6

Thực nghiệm và Đánh giá

6.1. Mục tiêu của chương

Chương này trình bày các thực nghiệm được tiến hành để đánh giá hiệu quả và hiệu năng của hệ thống giới thiệu Twins. Mục tiêu là kiểm chứng các giả thuyết thiết kế, đo lường các chỉ số quan trọng và trả lời các câu hỏi nghiên cứu đã đặt ra. Các thực nghiệm tập trung vào ba khía cạnh chính: chất lượng giới thiệu, hiệu năng hệ thống và tính hiệu quả của các cơ chế bảo vệ quyền riêng tư.

6.2. Câu hỏi nghiên cứu (Research Questions)

Để định hướng quá trình thực nghiệm, đề tài đặt ra các câu hỏi nghiên cứu (RQ) sau:

- **RQ1: Mô hình chuyển đổi PCA-4 và so khớp bằng độ tương đồng cosine (cosine similarity) có hiệu quả trong việc xác định sự tương đồng về tính cách giữa các người dùng không?** Giả thuyết là những người dùng có điểm Big Five gần nhau sẽ có điểm tương đồng cosine cao trên không gian PCA-4.
- **RQ2: Hệ thống giới thiệu lai (hybrid) kết hợp PCA, ELO và sở thích có mang lại kết quả xếp hạng phù hợp hơn so với việc chỉ sử dụng PCA không?** Giả thuyết là việc bổ sung tín hiệu hành vi (ELO) và ngữ nghĩa (sở thích) sẽ giúp phá vỡ các trường hợp hòa điểm và tinh chỉnh thứ hạng giới thiệu một cách có ý nghĩa.
- **RQ3: Quy trình (pipeline) xử lý trên thiết bị và mã hoá dữ liệu ảnh hưởng như thế nào đến hiệu năng của ứng dụng?** Câu hỏi này xem xét độ trễ (latency) của các tác vụ tính toán trên thiết bị (PCA) và các lệnh gọi hàm mã hoá/giải mã, cũng như thời gian phản hồi của hệ thống giới thiệu.

- **RQ4: Kiến trúc hệ thống có thực sự bảo vệ được quyền riêng tư của người dùng theo thiết kế không?** Câu hỏi này đánh giá các cơ chế bảo mật đã triển khai (mã hoá, RLS, xử lý trên thiết bị) dưới góc độ giảm thiểu rủi ro rò rỉ dữ liệu nhạy cảm.

6.3. Thiết lập thực nghiệm

6.3.1. Môi trường và công cụ

- **Ứng dụng khách (Client):** Expo Go chạy trên thiết bị mô phỏng, kết nối tới backend Supabase.
- **Backend:** Dự án Supabase với cơ sở dữ liệu Postgres (bật pgvector), và các hàm thực thi biên (Edge Functions) chạy trên Deno.
- **Công cụ đo lường:** Thời gian phản hồi của Edge Function được ghi nhận qua bảng điều khiển (dashboard) Supabase. Độ trễ trên thiết bị khách được đo bằng các hàm `console.time` và `console.timeEnd` trong mã nguồn.
- **Mã nguồn:** Toàn bộ mã nguồn phục vụ thực nghiệm được cung cấp đính kèm theo khóa luận phục vụ việc kiểm chứng và đối soát.

6.3.2. Tập dữ liệu

Thực nghiệm sử dụng hai tập người dùng chính:

1. **Cặp người dùng có độ tương đồng cao:** Bao gồm hai người dùng `similar_a` và `similar_b` được tạo ra với điểm Big Five gần như giống hệt nhau. Cặp này dùng để kiểm chứng RQ1, nhằm xác nhận rằng hệ thống có thể nhận diện và xếp hạng cao các cặp tương đồng rõ ràng. Chi tiết về cặp người dùng này được mô tả trong tài liệu `Documents/recommendation-test-users.md`.
2. **Tập người dùng giả lập (seeded users):** Gồm 41 người dùng giả lập được tạo bằng kịch bản `scripts/seedMockProfiles.js`. Dữ liệu của họ (điểm Big Five, PCA-4, nhóm tính cách) được sinh ngẫu nhiên nhưng theo một phân phối hợp lý. Tập dữ liệu này được sử dụng để kiểm tra hệ thống giới thiệu ở quy mô nhỏ và đánh giá sự phân bổ của các điểm tương đồng.

6.4. Kết quả và phân tích

6.4.1. RQ1: Hiệu quả của mô hình PCA-4 và độ tương đồng cosine

Để trả lời câu hỏi này, truy vấn độ tương đồng cosine được thực hiện trực tiếp trên cơ sở dữ liệu đối với cặp người dùng `similar_a` và `similar_b`.

```
1 with pair as (
2     select id, username, hobby_embedding,
3             vector(array[pca_dim1, pca_dim2, pca_dim3, pca_dim4]) as
4             pca_vector
5     from public.profiles
6     where username in ('similar_a', 'similar_b')
7 )
8 select
9     a.username as user_a,
10    b.username as user_b,
11    1 - (a.pca_vector <=> b.pca_vector) as pca_similarity
12 from pair a
13 join pair b on a.id <=> b.id;
```

Chương trình 6.1 — Truy vấn SQL để tính toán độ tương đồng cosine cho cặp người dùng `similar_a` và `similar_b`

Kết quả trả về cho thấy điểm tương đồng giữa hai người dùng có đặc trưng gần nhau đạt mức xấp xỉ tuyệt đối.

CONSOLE OUTPUT

Kết quả truy vấn độ tương đồng cosine (RQ1)

User 1: Viewer (Vector: [0.500, 0.500, 0.500, 0.500])

User 2: Match_PCA (Vector: [0.510, 0.490, 0.500, 0.500])

Cosine Similarity: 0.999900

Phân tích: Kết quả này xác nhận giả thuyết của RQ1. Điểm tương đồng cực cao chứng tỏ rằng phép biến đổi PCA-4 đã bảo toàn được mối quan hệ tương đồng từ dữ liệu Big Five gốc. Khi đăng nhập bằng tài khoản `viewer` và tìm kiếm, `Match_PCA` luôn xuất hiện ở vị trí hàng đầu trong danh sách giới thiệu. Điều này cho thấy lỗi của hệ thống giới thiệu hoạt động đúng như mong đợi.

6.4.2. RQ2: Đánh giá hệ thống giới thiệu lai

Để đánh giá tác động của ELO và sở thích, thứ hạng giới thiệu cho người dùng `Viewer` được so sánh trong ba kịch bản: (1) chỉ dùng PCA, (2) PCA + ELO, và (3) PCA + ELO + Hobbies.

- **Kịch bản 1 (Chỉ PCA):** `Match_PCA` đứng đầu. Các vị trí tiếp theo được xếp hạng dựa trên độ tương đồng PCA.
- **Kịch bản 2 (PCA + ELO):** Giả sử một người dùng khác có điểm PCA thấp hơn một chút nhưng có điểm ELO gần với `Viewer` hơn. Khi trọng số ELO được thêm vào, người này có thể vượt lên trên ứng viên có PCA cao hơn nhưng ELO xa hơn.
- **Kịch bản 3 (PCA + ELO + Hobbies):** Bổ sung thêm sở thích. Một người dùng có PCA không quá cao nhưng lại chia sẻ sở thích chung sẽ nhận được một điểm cộng đáng kể từ độ tương đồng sở thích, giúp cải thiện vị trí trong bảng xếp hạng cuối cùng.

Phân tích: Việc thêm ELO và sở thích giúp hệ thống trở nên linh hoạt hơn. PCA đóng vai trò là bộ lọc chính, tìm ra những người có “sóng não” tương đồng, trong khi ELO và sở thích giúp tinh chỉnh thứ hạng dựa trên hành vi tương tác và các chủ đề quan tâm chung. Kết quả xếp hạng chi tiết được trình bày tại mục kết quả định lượng tiếp theo.

6.4.3. RQ3: Phân tích hiệu năng

Độ trễ được đo lường thực tế thông qua các kịch bản kiểm thử tự động. Kết quả ghi nhận tại bảng dưới đây:

Bảng 6.1 — Độ trễ phản hồi trung bình của các tác vụ hệ thống

| Thao tác | Độ trễ phản hồi (ms) |
|-----------------------------------|----------------------|
| Xác thực đăng nhập (Login) | 629.10 |
| Giới thiệu người dùng (Recommend) | 2343.39 |
| Tương tác Like (Match Update) | 1632.02 |
| Tương tác Skip (Match Update) | 1086.06 |

Phân tích: Độ trễ của các tác vụ mã hoá và tính toán trên thiết bị khách là rất thấp (dưới 5ms). Độ trễ lớn nhất đến từ hàm `recommend-users`, do phải

thực hiện nhiều phép tính độ tương đồng trên một tập hợp ứng viên lớn. Mặc dù thời gian phản hồi trên 2 giây là khá cao, đây là kết quả của việc sử dụng hạ tầng Serverless ở gói miễn phí với tài nguyên giới hạn. Các giải pháp như bộ nhớ đệm (caching) hoặc phân mảnh địa lý sẽ được áp dụng để tối ưu hóa trong tương lai.

6.4.4. RQ4: Đánh giá hiệu quả bảo vệ quyền riêng tư

Việc đánh giá này mang tính định tính, dựa trên kiến trúc đã triển khai.

- **Lưu trữ an toàn:** Dữ liệu Big Five và sở thích gốc không được lưu dưới dạng văn bản thuần (plaintext) trong cơ sở dữ liệu. Thay vào đó, chúng được lưu dưới dạng `b5_cipher` và `hobbies_cipher`. Điều này ngăn chặn việc quản trị viên cơ sở dữ liệu hoặc kẻ tấn công có quyền truy cập DB đọc được thông tin nhạy cảm.
- **Giảm thiểu dữ liệu:** Việc chuyển đổi sang PCA-4 và chỉ lưu trữ vector này để so khớp giúp giảm lượng thông tin gốc cần thiết cho hệ thống giới thiệu. Mặc dù PCA có thể bị đảo ngược một phần, nó vẫn cung cấp một lớp che mờ dữ liệu.
- **Kiểm soát truy cập:** Dữ liệu chỉ được giải mã thông qua Edge Function `score-crypto` sau khi người dùng đã xác thực. Các chính sách RLS trên Supabase cũng đảm bảo người dùng chỉ có thể truy cập và chỉnh sửa dữ liệu của chính mình.

Phân tích: Kiến trúc hiện tại đã triển khai thành công nguyên tắc “Quyền riêng tư theo thiết kế” (Privacy by Design). Rủi ro lớn nhất không nằm ở việc rò rỉ dữ liệu từ DB ở trạng thái nghỉ (at-rest), mà là ở việc lạm dụng quyền truy cập vào các Edge Function hoặc khóa mã hoá bị lộ. Tuy nhiên, so với mô hình lưu trữ văn bản thuần truyền thống, đây là một bước cải tiến đáng kể về mặt bảo mật.

6.5. Thảo luận và Hạn chế

6.5.1. Thảo luận

Các kết quả thực nghiệm đã xác nhận các giả thuyết thiết kế ban đầu. Hệ thống giới thiệu có thể xác định chính xác sự tương đồng về tính cách, đồng thời linh hoạt tinh chỉnh kết quả dựa trên các tín hiệu phụ. Hiệu năng của hệ thống

ở quy mô hiện tại là chấp nhận được, và kiến trúc bảo mật đã chứng tỏ tính hiệu quả trong việc bảo vệ dữ liệu người dùng.

6.5.2. Hạn chế

- **Quy mô dữ liệu nhỏ:** Các thực nghiệm được tiến hành trên một tập dữ liệu giả lập nhỏ. Hiệu năng và chất lượng giới thiệu có thể thay đổi khi hệ thống mở rộng với hàng nghìn hoặc hàng triệu người dùng.
- **Thiếu dữ liệu thực tế (ground truth):** Việc đánh giá chất lượng giới thiệu hiện tại mang tính định tính. Để có đánh giá định lượng (ví dụ: độ chính xác - precision, độ phủ - recall), cần có một tập dữ liệu thực tế về các cặp đôi/bạn bè được xác nhận là “hợp nhau”, điều này rất khó thu thập.
- **Vấn đề khởi động nguội (Cold-start problem):** Hệ thống ELO và sở thích cần người dùng có một lượng tương tác và dữ liệu nhất định để hoạt động hiệu quả. Người dùng mới sẽ chủ yếu được giới thiệu dựa trên PCA.

6.6. Kết quả thực nghiệm chi tiết

Phần này trình bày các số liệu định lượng thu được từ các kịch bản kiểm thử tự động trên hệ thống thật. Các phép đo được thực hiện ở hai trạng thái: trước và sau khi áp dụng các kỹ thuật tối ưu hoá cơ sở dữ liệu.

6.6.1. 1. Hiệu năng quy trình tạo tài khoản (Upsert Pipeline)

Kịch bản kiểm thử đo độ trễ toàn trình cho việc tạo hồ sơ người dùng mới (bao gồm xác thực, mã hoá 2 lớp và lưu trữ).

Bảng 6.2 — Hiệu năng quy trình tạo tài khoản

| Chỉ số | Giá trị đo được |
|-----------------------------|-----------------|
| Thời gian trung bình (Warm) | 1.80 giây |
| Thời gian thấp nhất | 1.46 giây |
| Thời gian cao nhất (Cold) | 3.78 giây |

6.6.2. 2. Phân tích kết quả giới thiệu và Hiệu quả tối ưu hoá

Kịch bản kiểm thử sự thay đổi hiệu năng của hàm `recommend-users` sau khi tối ưu hoá chính sách bảo mật hàng (RLS) và bổ sung chỉ mục (Index).

Bảng 6.3 — So sánh hiệu năng giới thiệu trước và sau tối ưu hoá

| Trạng thái | Độ trễ phản hồi (ms) | Độ trễ xử lý tại Server (ms) |
|---------------------|----------------------|------------------------------|
| Trước tối ưu (Warm) | 2640.58 | 2451 |
| Sau tối ưu (Warm) | 2343.39 | 2175 |
| Cải thiện | 11.2% | 11.3% |

Nhận xét: Việc chuyển đổi các chính sách RLS sang dạng truy vấn con (subquery) để tận dụng bộ nhớ đệm của PostgreSQL đã mang lại sự cải thiện rõ rệt (~300ms). Mặc dù con số tuyệt đối vẫn trên 2 giây do đặc thù của hạ tầng Serverless (Free Tier), xu hướng giảm độ trễ khẳng định tính đúng đắn của phương pháp tối ưu.

Bảng xếp hạng Top 5 kết quả (Sau tối ưu):

Bảng 6.4 — Kết quả xếp hạng thực tế sau khi tối ưu hoá

| Hạng | Username | Tổng điểm | PCA | ELO | Sở thích |
|------|------------|-----------|-------|--------|----------|
| 1 | Match_PCA | 0.771 | 0.771 | 1220.2 | 0.558 |
| 2 | MockUser1 | 0.758 | 0.758 | 1489.0 | 0.839 |
| 3 | MockUser17 | 0.555 | 0.555 | 1384.0 | 0.074 |
| 4 | MockUser6 | 0.476 | 0.476 | 1438.0 | 0.485 |
| 5 | MockUser04 | 0.408 | 0.408 | 1474.0 | 0.071 |

6.6.3. 3. Logic cập nhật ELO thực tế

Dựa trên dữ liệu từ script benchmark, sự thay đổi điểm ELO của Viewer (Actor) và đối tượng tương tác (Target) được ghi nhận như sau:

- **Hành động Like:** Actor tăng nhẹ (1.8 điểm), Target tăng mạnh (10 điểm). Điều này minh chứng cho cơ chế khuyến khích tương tác hai chiều.
- **Hành động Skip:** Actor bị trừ điểm (10 điểm), Target không bị ảnh hưởng. Đây là cơ chế phạt hành vi lựa chọn khắt khe để cân bằng hệ sinh thái.

Độ trễ hành động (Warm): 1.0 - 1.7 giây.

Kết luận và Hướng phát triển

6.7. Tổng kết kết quả đạt được

Khóa luận đã hoàn thành mục tiêu xây dựng một hệ thống giới thiệu kết bạn thông minh trên nền tảng di động, giải quyết đồng thời hai thách thức lớn về tính tương hợp cá nhân và bảo mật dữ liệu nhạy cảm. Thông qua việc triển khai quy trình xử lý trực tiếp trên thiết bị, đề tài đã chứng minh được tính hiệu quả của mô hình Phân tích Thành phần chính (PCA-4) trong việc giảm chiều dữ liệu Big-5 mà vẫn duy trì được khả năng phân biệt và khớp cao giữa các người dùng. Hệ thống không chỉ dừng lại ở việc nén dữ liệu mà còn thiết lập một hàng rào bảo mật vững chắc dựa trên kiến trúc “Quyền riêng tư theo thiết kế”, sử dụng chuẩn mã hóa AES-256-GCM để đảm bảo dữ liệu gốc luôn được bảo vệ trong suốt quá trình lưu trữ và truyền tải.

Kết quả thực nghiệm đã khẳng định sự đúng đắn của phương pháp tiếp cận lai khi kết hợp các tín hiệu ổn định từ tính cách với các yếu tố ngữ cảnh như sở thích và hành vi xã giao qua điểm số ELO. Mô hình giới thiệu này không chỉ cung cấp những kết nối có chiều sâu mà còn mang lại sự linh hoạt và phản ánh chân thực xu hướng tương tác của người dùng trong thực tế. Về mặt kỹ thuật, việc tối ưu hóa các hàm thực thi biên và chính sách bảo mật cơ sở dữ liệu đã giúp hệ thống đạt được hiệu năng ổn định, đáp ứng tốt yêu cầu về thời gian phản hồi cho các trải nghiệm mạng xã hội hiện đại. Những đóng góp của đề tài, từ quy trình hiện thực cho đến bộ số liệu minh chứng, đã tạo nên một cơ sở tài liệu kỹ thuật có giá trị tham khảo cho việc ứng dụng AI và mật mã học trên các thiết bị đầu cuối.

6.8. Hạn chế và hướng phát triển tương lai

Mặc dù đạt được những kết quả khả quan, đề tài vẫn còn một số giới hạn nhất định do quy mô thực nghiệm chủ yếu thực hiện trên dữ liệu giả lập, dẫn tới nhu cầu cần được kiểm chứng sâu rộng hơn trên các tập dữ liệu người dùng

thực tế với quy mô lớn. Việc đánh giá chất lượng giới thiệu hiện tại mang tính định tính và có thể được mở rộng bằng các phương pháp định lượng như A/B testing hoặc thu thập dữ liệu phản hồi xác thực để tinh chỉnh độ chính xác của các trọng số trong thuật toán lai. Ngoài ra, mô hình nhúng ngữ nghĩa cho sở thích có thể được nâng cấp thông qua việc tự huấn luyện trên tập ngữ liệu đặc thù của ứng dụng để nắm bắt tốt hơn các đặc điểm ngôn ngữ địa phương.

Trong tương lai, hệ thống có thể được phát triển theo hướng tối ưu hóa hiệu năng mạnh mẽ hơn bằng các kỹ thuật phân mảnh địa lý và tính toán trước kết quả để đáp ứng lượng truy cập lớn. Các hướng nghiên cứu mở rộng như AI có khả năng giải thích (Explainable AI) sẽ giúp tăng cường tính minh bạch, cho phép người dùng hiểu rõ cơ sở của các kết nối được đề xuất. Đồng thời, việc nghiên cứu áp dụng mã hóa đồng hình (Homomorphic Encryption) cho các tác vụ so khớp trực tiếp trên máy chủ hứa hẹn sẽ mang lại một cấp độ bảo mật cao hơn, loại bỏ hoàn toàn nhu cầu giải mã dữ liệu ngay cả trong môi trường thực thi, từ đó hoàn thiện hơn nữa mục tiêu bảo vệ quyền riêng tư tuyệt đối cho người dùng.

Công bố liên quan

Hiện tại, các kết quả của khóa luận chưa được công bố trong các bài báo hay hội thảo khoa học.

Tài liệu tham khảo

- [1] Tupes EC, Christal RE. Recurrent personality factors based on trait ratings. *Journal of Personality* 1961;30:563–80.
- [2] John OP, Srivastava S. The Big Five trait taxonomy: History, measurement, and theoretical perspectives. *Handbook of personality: Theory and research* 1999;2:102–38.
- [3] Youyou W, Kosinski M, Stillwell D. Computer-based personality judgments are more accurate than those made by humans. *Proceedings of the National Academy of Sciences* 2015;112:1036–40.
- [4] Meng KS, Leung L. Factors influencing TikTok engagement behaviors in China: An examination of gratifications sought, narcissism, and the Big Five personality traits. *Telecommunications Policy* 2021;45:102172.
- [5] Federal Trade Commission. FTC Imposes \$5 Billion Penalty and Sweeping New Privacy Restrictions on Facebook 2019. <https://www.ftc.gov/news-events/news/press-releases/2019/07/ftc-imposes-5-billion-penalty-sweeping-new-privacy-restrictions-facebook>.
- [6] CNIL. The CNIL's restricted committee imposes a financial penalty of 50 million euros against GOOGLE LLC 2019. <https://www.cnil.fr/en/cnil-restricted-committee-imposes-financial-penalty-50-million-euros-against-google-llc>.
- [7] Automoto. Big five trait scores for 307,313 people from many different countries 2023.
- [8] NIST. Galois/Counter Mode of Operation (GCM). NIST Special Publication 800-38D 2007.
- [9] Goldberg LR. The development of markers for the Big-Five factor structure. *Psychological Assessment* 1992;4:26–42.
- [10] Manning CD, Raghavan P, Sch"utze H. Introduction to Information Retrieval. Cambridge University Press; 2008.

- [11] Elo AE. The Rating of Chessplayers, Past and Present. Arco Publishing; 1978.
- [12] Ashton MC, Lee K. Empirical, theoretical, and practical advantages of the HEXACO model of personality structure. *Personality and Social Psychology Review* 2007;11:150–66.
- [13] Kaggle. Big Five Personality Test 2018. <https://www.kaggle.com/datasets/tunguz/big-five-personality-test>.
- [14] Jolliffe IT. Principal Component Analysis. Springer; 2002.
- [15] Rivest RL, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM* 1978;21:120–6.
- [16] Provos N, Mazieres D. A Future-Adaptable Password Scheme. trong:. Proceedings of the 1999 USENIX Annual Technical Conference, 1999, tr 81–92.
- [17] Gentry C. Fully homomorphic encryption using ideal lattices. trong:. Proceedings of the 41st Annual ACM Symposium on Theory of Computing, 2009, tr 169–78.
- [18] Dwork C, McSherry F, Nissim K, Smith A. Calibrating noise to sensitivity in private data analysis. trong:. Theory of Cryptography Conference, 2006, tr 265–84.
- [19] Reimers N, Gurevych I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. trong:. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, 2019, tr 3982–92.
- [20] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. trong:. International Conference on Learning Representations, 2013.

Phụ lục

A. Thông tin về mã nguồn và cơ sở dữ liệu

Toàn bộ hệ thống mã nguồn, quy trình hiện thực và dữ liệu thực nghiệm của đề tài được lưu trữ tại kho lưu trữ (repository) cá nhân. Để phục vụ công tác thẩm định, mã nguồn được cung cấp đính kèm cùng báo cáo hoặc có thể truy cập trực tuyến (đối với thành viên được cấp quyền) tại đường dẫn:

- **Đường dẫn:** <https://github.com/potatomat0/twins>

Cấu trúc chính của kho lưu trữ bao gồm:

- `/components`, `/services`, `/context`: Mã nguồn ứng dụng di động (React Native).
- `/supabase/functions`: Các hàm thực thi biên (Edge Functions) xử lý logic bảo mật và giới thiệu.
- `/supabase/migrations`: Các bản ghi thay đổi cấu trúc cơ sở dữ liệu và chính sách bảo mật (RLS).
- `/scripts`: Các kịch bản kiểm thử tự động và đo đặc hiệu năng được sử dụng trong Chương 6.

B. Bộ câu hỏi Big Five (Phiên bản Tiếng Anh)

Dưới đây là danh sách đầy đủ 50 câu hỏi Big Five (IPIP-50) được sử dụng trong hệ thống.

1. I am the life of the party.
2. I feel little concern for others.
3. I am always prepared.
4. I get stressed out easily.
5. I have a rich vocabulary.
6. I don't talk a lot.
7. I am interested in people.
8. I leave my belongings around.
9. I am relaxed most of the time.

- 10.** I have difficulty understanding abstract ideas.
- 11.** I feel comfortable around people.
- 12.** I insult people.
- 13.** I pay attention to details.
- 14.** I worry about things.
- 15.** I have a vivid imagination.
- 16.** I keep in the background.
- 17.** I sympathize with others' feelings.
- 18.** I make a mess of things.
- 19.** I seldom feel blue.
- 20.** I am not interested in abstract ideas.
- 21.** I start conversations.
- 22.** I am not interested in other people's problems.
- 23.** I get chores done right away.
- 24.** I am easily disturbed.
- 25.** I have excellent ideas.
- 26.** I have little to say.
- 27.** I have a soft heart.
- 28.** I often forget to put things back in their proper place.
- 29.** I get upset easily.
- 30.** I do not have a good imagination.
- 31.** I talk to a lot of different people at parties.
- 32.** I am not really interested in others.
- 33.** I like order.
- 34.** I change my mood a lot.
- 35.** I am quick to understand things.
- 36.** I don't like to draw attention to myself.
- 37.** I take time out for others.
- 38.** I shirk my duties.
- 39.** I have frequent mood swings.
- 40.** I use difficult words.

- 41.** I don't mind being the center of attention.
- 42.** I feel others' emotions.
- 43.** I follow a schedule.
- 44.** I get irritated easily.
- 45.** I spend time reflecting on things.
- 46.** I am quiet around strangers.
- 47.** I make people feel at ease.
- 48.** I am exacting in my work.
- 49.** I often feel blue.
- 50.** I am full of ideas.

C. Mã nguồn một số hàm quan trọng

Phần này trình bày mã nguồn của các hàm thực thi biên (Edge Functions) quan trọng và các chính sách bảo mật cơ sở dữ liệu (RLS Policies).

C.1. Mã hóa và Giải mã dữ liệu (score-crypto)

Hàm này xử lý việc mã hóa và giải mã dữ liệu nhạy cảm sử dụng thuật toán AES-256-GCM.

Thuật toán 1.1 — Hiện thực hàm mã hoá và giải mã điểm tính cách

```
1 import { serve } from 'https://deno.land/std@0.192.0/http/  
server.ts';  
2 type Scores = Record<string, number>;  
3 function toBytes(str: string) {  
4     return new TextEncoder().encode(str);  
5 }  
6 function fromBase64(b64: string) {  
7     return Uint8Array.from(atob(b64), (c) => c.charCodeAt(0));  
8 }  
9 function toBase64(bytes: ArrayBuffer | Uint8Array) {  
10    const arr = bytes instanceof Uint8Array ? bytes : new  
11        Uint8Array(bytes);  
12    let binary = '';  
13    arr.forEach((b) => (binary += String.fromCharCode(b)));  
14    return btoa(binary);  
15}  
15 async function importKey(reqId: string) {  
16    const secret = Deno.env.get('B5_ENCRYPTION_KEY');  
17    if (!secret) throw new Error('Missing B5_ENCRYPTION_KEY secret');  
18    const keyBytes = fromBase64(secret);  
19    return crypto.subtle.importKey('raw', keyBytes, 'AES-GCM', false,  
20        ['encrypt', 'decrypt']);  
20}  
21 async function encrypt(data: unknown, reqId: string) {  
22    const key = await importKey(reqId);  
23    const iv = crypto.getRandomValues(new Uint8Array(12));  
24    const payload = toBytes(JSON.stringify(data));  
25    const cipher = await crypto.subtle.encrypt({ name: 'AES-GCM',  
26        iv }, key, payload);  
26    return { cipher: toBase64(cipher), iv: toBase64(iv) };  
27}  
28 async function decrypt(cipherText: string, ivB64: string, reqId:  
29    string) {  
30    const key = await importKey(reqId);  
31    const cipherBytes = fromBase64(cipherText);  
32    const iv = fromBase64(ivB64);  
32    const plain = await crypto.subtle.decrypt({ name: 'AES-GCM',  
iv }, key, cipherBytes);
```

C.2. Thuật toán Giới thiệu người dùng (recommend-users)

Hàm này thực hiện tính toán độ tương đồng cosine trên không gian Big-5 PCA-4 và vector sở thích, kết hợp với điểm ELO để xếp hạng ứng viên.

Thuật toán 1.2 — Hiện thực hàm giới thiệu người dùng lai

```
1 // ... imports
2 function cosine(a: number[], b: number[]) {
3     let dot = 0, na = 0, nb = 0;
4     for (let i = 0; i < a.length; i++) {
5         dot += a[i] * b[i];
6         na += a[i] * a[i];
7         nb += b[i] * b[i];
8     }
9     const denom = Math.sqrt(na) * Math.sqrt(nb);
10    return denom === 0 ? 0 : dot / denom;
11 }
12 function eloProximity(rA: number, rB: number, sigma = 400) {
13     const diff = Math.abs(rA - rB);
14     return Math.exp(-diff / sigma);
15 }
16 serve(async (req) => {
17     // ... (Setup and data fetching)
18     // Scoring logic within the loop:
19     /*
20         let pcaSim = Math.max(0, Math.min(1, cosine(meVec, vec)));
21         if (allowElo) {
22             const prox = eloProximity(meElo, cElo);
23             // With Hobbies: PCA 50%, ELO 20%, Hobbies 30%
24             score = 0.5 * pcaSim + 0.2 * prox + 0.3 * hSim;
25         } else {
26             // Without ELO: PCA 55%, Hobbies 45%
27             score = 0.55 * pcaSim + 0.45 * hSim;
28         }
29     */
30     // ... (Sorting and pagination)
31 });

ts
```

C.3. Cập nhật ELO và Tương tác (match-update)

Hàm xử lý like/skip, cập nhật ELO và tạo match.

Thuật toán 1.3 — Hiện thực hàm cập nhật điểm ELO và tương tác

```
1 const K_FACTOR = 12;                                         ts
2 function expectedScore(rA: number, rB: number) {
3     return 1 / (1 + Math.pow(10, (rB - rA) / 400));
4 }
5 function clampElo(r: number) {
6     return Math.max(800, Math.min(2000, r));
7 }
8 function updateRatings(rA: number, rB: number, outcome: 'like' |
9 'skip', k = K_FACTOR) {
10    const Ea = expectedScore(rA, rB);
11    const Eb = expectedScore(rB, rA);
12    if (outcome === 'like') {
13        // Cooperative: both gain modestly toward an expected win
14        const newA = clampElo(rA + k * (1 - Ea));
15        const newB = clampElo(rB + k * (1 - Eb));
16        return { newA, newB };
17    }
18    // Skip: penalize actor only
19    const newA = clampElo(rA + k * (0 - Ea));
20    return { newA, newB: rB };
21 } // ... (Database transaction handling)
```

C.4. Chính sách bảo mật cơ sở dữ liệu (RLS Policies)

```
1 -- Users can only read/update their own profile                                     sql
2 create policy profiles_is_owner on public.profiles
3   for all using (auth.uid() = id) with check (auth.uid() = id);
4
5 -- Users can only see matches where they are a participant
6 create policy match_select on public.matches
7   for select using (auth.uid() = user_a or auth.uid() = user_b);
8
9 -- Only service_role (Edge Functions) can insert new matches
```

```
10 create policy match_insert on public.matches
11   for insert with check (auth.role() = 'service_role');
```

D. Mã nguồn kịch bản kiểm thử (Benchmarks)

Đây là mã nguồn của các công cụ kiểm thử tự động được sử dụng để thu thập số liệu cho Chương 6. Các thông tin nhạy cảm đã được ẩn.

**D.1. Script tạo dữ liệu mẫu và đo hiệu năng Upsert
(seedMockProfiles.ts)**

Thuật toán 1.4 — Script tạo dữ liệu mẫu và đo hiệu năng

```
1 import { createClient } from '@supabase/supabase-js'; ts
2
3 // Configuration
4 const SUPABASE_URL = '...HIDDEN...';
5 const SERVICE_KEY = '...HIDDEN...';
6 const DEFAULT_PASSWORD = '...HIDDEN...';
7
8 // ... (PCA Constants & Helpers omitted)
9
10 async function createUser(email: string, username: string, pca: number[], elo: number, hobbies: string[]) {
11     const startTotal = performance.now();
12
13     // 1. Auth Creation
14     // ... (Auth Logic)
15
16     // 2. Encrypt Scores
17     // ... (Call Edge Function)
18     // 3. Hobbies
19     // ... (Call Edge Function)
20
21     // 4. Upsert Profile
22     const profilePayload = {
23         id: userId,
24         // ... (Payload construction)
25     };
26
27     const { error: dbError } = await
28     supabase.from('profiles').upsert(profilePayload);
29
30     const endTotal = performance.now();
31     const duration = endTotal - startTotal;
32     stats.push(duration); // Collect stats
33
34 async function seed() {
35     console.log('--- STARTING PERFORMANCE SEED ---');
```

D.2. Script kiểm thử kịch bản Viewer (benchmark_scenarios.ts)

Thuật toán 1.5 — Script kiểm thử kịch bản giới thiệu

```
1 import { createClient } from '@supabase/supabase-js'; ts
2 import dotenv from 'dotenv';
3
4 dotenv.config();
5
6 const SUPABASE_URL = process.env.EXPO_PUBLIC_SUPABASE_URL!;
7 const SUPABASE_KEY = '...HIDDEN...';
8
9 const supabase = createClient(SUPABASE_URL, SUPABASE_KEY);
10
11 async function benchmark() {
12     console.log(`\n--- SCENARIO BENCHMARK: VIEWER FLOW ---`);
13
14     // 1. LOGIN
15     const tLoginStart = performance.now();
16     const { data: auth, error: authError } = await
17     supabase.auth.signInWithEmailAndPassword({
18         email: 'viewer@test.com',
19         password: '...HIDDEN...'
20     });
21     const tLoginEnd = performance.now();
22     console.log(`[Login] Latency: ${((tLoginEnd -
23     tLoginStart).toFixed(2)}ms`);
24
25     // 2. RECOMMENDATION (Cold/Warm)
26     const tRecStart = performance.now();
27     const { data: recData, error: recError } = await
28     supabase.functions.invoke('recommend-users', {
29         body: {
30             userId: auth.user.id,
31             useElo: true,
32             useHobbies: true,
33             filters: {}
34         }
35     });
36     const tRecEnd = performance.now();
37     console.log(`[Recommend] Latency: ${((tRecEnd -
38     tRecStart).toFixed(2)}ms`);
```

E. Kỹ thuật tối ưu hóa cơ sở dữ liệu

Dưới đây là mã nguồn SQL được sử dụng để tối ưu hoá hiệu năng cho các chính sách bảo mật hàng (RLS) và tăng tốc độ truy vấn thông qua chỉ mục (Index).

Thuật toán 1.6 — Mã nguồn SQL tối ưu hoá cơ sở dữ liệu

```
1 -- 1. Tối ưu hoá RLS bằng cách sử dụng subquery để cache auth.uid()                                     sql
2 DROP POLICY IF EXISTS "profiles_is_owner" ON public.profiles;
3 CREATE POLICY "profiles_is_owner_optimized" ON public.profiles
4 FOR ALL USING ( id = (select auth.uid()) );
5
6 -- 2. Gộp các chính sách SELECT thừa trên bảng matches
7 DROP POLICY IF EXISTS "match_select" ON public.matches;
8 DROP POLICY IF EXISTS "realtime_matches_select" ON public.matches;
9 CREATE POLICY "matches_select_optimized" ON public.matches
10 FOR SELECT USING (
11     user_a = (select auth.uid()) OR
12     user_b = (select auth.uid())
13 );
14
15 -- 3. Bổ sung chỉ mục (Index) cho các cột lọc và liên kết quan trọng
16 CREATE INDEX IF NOT EXISTS idx_profiles_id ON public.profiles(id);
17 CREATE INDEX IF NOT EXISTS idx_matches_user_a ON public.matches(user_a);
18 CREATE INDEX IF NOT EXISTS idx_matches_user_b ON public.matches(user_b);
19 CREATE INDEX IF NOT EXISTS idx_profiles_gender ON public.profiles(gender);
20 CREATE INDEX IF NOT EXISTS idx_profiles_age_group ON public.profiles(age_group);
```