**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHƯƠNG TRÌNH CHẤT LƯỢNG CAO**

**NGUYỄN GIA HÂN - MẠCH CẢNH TOÀN**

**XÂY DỰNG KHO DỮ LIỆU ĐỒ THỊ DỰA TRÊN NỀN TẢNG DỮ LIỆU LỚN CHO ĐỀ XUẤT KHÓA HỌC DỰA TRÊN MỤC TIÊU NGHỀ NGHIỆP CNTT**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN CNTT**

**TP. HCM, 2023**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**CHƯƠNG TRÌNH CHẤT LƯỢNG CAO**

**NGUYỄN GIA HÂN - 19127134**

**MẠCH CẢNH TOÀN - 19127584**

**XÂY DỰNG KHO DỮ LIỆU ĐỒ THỊ DỰA TRÊN NỀN TẢNG DỮ LIỆU LỚN CHO ĐỀ XUẤT KHÓA HỌC DỰA TRÊN MỤC TIÊU NGHỀ NGHIỆP CNTT**

**KHÓA LUẬN TỐT NGHIỆP CỬ NHÂN CNTT**

**GIÁO VIÊN HƯỚNG DẪN**

**TS. NGUYỄN TRẦN MINH THƯ**

**TP. HCM, 2023**

# Lời cảm ơn

Lời đầu tiên chúng em xin phép gửi lời cảm ơn đến cô Nguyễn Trần Minh Thư đã hướng dẫn và giúp đỡ chúng em tận tình trong suốt quá trình nghiên cứu và thực hiện khóa luận tốt nghiệp này. Trong suốt quá trình nghiên cứu, những góp ý, chia sẻ và nhắc nhở từ cô đã góp phần quan trọng trong việc giúp chúng em có thể hoàn thiện hơn và vượt qua những khó khăn trong khóa luận này. Chúng em cũng xin gửi lời cảm ơn tới chị Vũ Yến Ngọc và anh Trần Văn Hiếu đã nhiệt tình hỗ trợ trong suốt quá trình thực hiện đề tài.

Đồng thời, chúng em cũng gửi làm cảm ơn sâu sắc tới quý thầy cô trường Đại học Khoa Học Tự Nhiên – Đại học Quốc gia TP.HCM, đặc biệt là quý thầy cô Khoa Công nghệ thông tin đã tận tình giảng dạy cho chúng em trong thời gian học tập tại trường. Chúng em được thầy cô truyền đạt các kiến thức quý báu, là nền tảng cho việc nghiên cứu khóa luận tốt nghiệp, và cũng là hành trang cho bản thân trong tương lai.

Do sự hạn chế về mặt kiến thức và thời gian thực hiện khóa luận nên các vấn đề gặp phải trong việc nghiên cứu khóa luận vẫn có thể chưa giải quyết được đầy đủ. Vì vậy, chúng em mong muốn được nhận được sự góp ý từ quý thầy cô để bài khóa luận tốt nghiệp được hoàn thiện hơn.

Lời cuối cùng, chúng em xin gửi lời cảm ơn tới quý thầy cô, kính chúc quý thầy cô thật nhiều sức khỏe, thành công và hạnh phúc!

Thành phố Hồ Chí Minh, ngày … tháng … năm 2023

**Nguyễn Gia Hân**

**Mạch Cảnh Toàn**

# Mục lục

[Lời cảm ơn i](#_Toc130545824)

[Mục lục ii](#_Toc130545825)

[Danh sách hình v](#_Toc130545826)

[Danh sách bảng viii](#_Toc130545827)

[Danh sách từ viết tắt x](#_Toc130545828)

[Tóm tắt xii](#_Toc130545829)

[Mở đầu 1](#_Toc130545830)

[1.1 Giới thiệu chung 1](#_Toc130545831)

[1.2 Mục tiêu đề tài 2](#_Toc130545832)

[1.3 Hướng tiếp cận đề tài 4](#_Toc130545833)

[1.4 Đóng góp của đề tài 5](#_Toc130545834)

[1.5 Bố cục trình bày 5](#_Toc130545835)

[Tổng quan về kho dữ liệu và kho dữ liệu đồ thị 7](#_Toc130545836)

[2.1 Kho dữ liệu và tình trạng của kho dữ liệu hiện nay 7](#_Toc130545837)

[2.1.1 Kho dữ liệu 7](#_Toc130545838)

[2.1.2 Tình trạng của kho dữ liệu hiện nay 8](#_Toc130545839)

[2.2 Kho dữ liệu theo nhóm chức năng 13](#_Toc130545840)

[2.2.1 Giới thiệu kho dữ liệu theo nhóm chức năng 13](#_Toc130545841)

[2.3 Kho dữ liệu đồ thị và ứng dụng 15](#_Toc130545842)

[2.3.1 Mô hình dữ liệu đồ thị 15](#_Toc130545843)

[2.3.2 Sơ lược về CSDL đồ thị 17](#_Toc130545844)

[2.3.3 Giới thiệu kho dữ liệu đồ thị và ứng dụng 18](#_Toc130545845)

[2.3.4 Khảo sát hiện trạng kho dữ liệu đồ thị 18](#_Toc130545846)

[2.3.5 Kết luận và so sánh kiến trúc của kho dữ liệu đồ thị và kho dữ liệu quan hệ 30](#_Toc130545847)

[2.4 Kết luận chương 33](#_Toc130545848)

[Kiến trúc dữ liệu đồ thị đề xuất 34](#_Toc130545849)

[3.1 Phạm vi bài toán 34](#_Toc130545850)

[3.2 Mô tả yêu cầu chức năng 35](#_Toc130545851)

[3.2.1 Nhóm chức năng phân tích thống kê 35](#_Toc130545852)

[3.2.2 Nhóm chức năng tìm kiếm 37](#_Toc130545853)

[3.2.3 Nhóm chức năng tư vấn 38](#_Toc130545854)

[3.3 Cơ sở thiết kế kiến trúc đồ thị đề xuất 40](#_Toc130545855)

[3.4 Kết luận chương 51](#_Toc130545856)

[Thiết kế kiến trúc tổng quát của hệ thống 52](#_Toc130545857)

[4.1 Thiết kế kiến trúc tổng quát của hệ thống 52](#_Toc130545858)

[4.2 Mô hình GDM trong kiến trúc của hệ thống 55](#_Toc130545859)

[4.2.1 Giải pháp cho nhóm chức năng phân tích thống kê 55](#_Toc130545860)

[4.2.2 Giải pháp cho nhóm chức năng tìm kiếm 58](#_Toc130545861)

[4.2.3 Giải pháp cho nhóm chức năng tư vấn 62](#_Toc130545862)

[4.3 Kết luận chương 66](#_Toc130545863)

[Phân tích lựa chọn công nghệ triển khai hệ thống 67](#_Toc130545864)

[5.1 Nhóm công nghệ trong kiến trúc triển khai 67](#_Toc130545865)

[5.2 Phân tích lựa chọn công nghệ 68](#_Toc130545866)

[5.2.1 Công nghệ lưu trữ CSDL đồ thị 68](#_Toc130545867)

[5.2.2 Công nghệ xử lý dữ liệu lớn 72](#_Toc130545868)

[5.2.3 Áp dụng công nghệ vào kiến trúc của hệ thống 74](#_Toc130545869)

[5.3 Kết luận chương 74](#_Toc130545870)

[Thiết kế triển khai và cài đặt thử nghiệm 75](#_Toc130545871)

[6.1 Triển khai hệ thống 75](#_Toc130545872)

[6.1.1 Hệ thống tập tin lưu trữ dữ liệu thô 77](#_Toc130545873)

[6.1.2 Sơ đồ xử lý dữ liệu và đổ dữ liệu vào kho dữ liệu đồ thị 78](#_Toc130545874)

[6.1.3 Sơ đồ xử lý chuyển dữ liệu từ GDW sang GDM 88](#_Toc130545875)

[6.1.3.1 Sơ đồ chuyển dữ liệu từ GDW sang SDM 88](#_Toc130545876)

[6.1.3.2 Sơ đồ chuyển dữ liệu từ GDW sang CDM 89](#_Toc130545877)

[6.1.3.3 Sơ đồ chuyển dữ liệu từ GDW sang UDM 91](#_Toc130545878)

[6.1.3.4 Sơ đồ chuyển dữ liệu từ GDW sang CCDM 91](#_Toc130545879)

[6.1.4 Kiến trúc đường ống dữ liệu 92](#_Toc130545880)

[6.2 Cài đặt hệ thống 93](#_Toc130545881)

[6.2.1 Thực hiện cài đặt 93](#_Toc130545882)

[6.2.2 Kết quả cài đặt hệ thống 96](#_Toc130545883)

[6.3 Thử nghiệm và phân tích kết quả 104](#_Toc130545884)

[6.3.1 Kịch bản thử nghiệm 104](#_Toc130545885)

[6.3.2 Đánh giá kết quả 110](#_Toc130545886)

[6.4 Kết luận chương 113](#_Toc130545887)

[Kết luận và hướng phát triển 114](#_Toc130545888)

[7.1 Kết luận 114](#_Toc130545889)

[7.1.1 Kết quả đạt được về mặt lý thuyết 114](#_Toc130545890)

[7.1.2 Kết quả đạt được về mặt triển khai hệ thống 115](#_Toc130545891)

[7.1.3 Các hạn chế 115](#_Toc130545892)

[7.2 Hướng phát triển 115](#_Toc130545893)

[Tài liệu tham khảo 117](#_Toc130545894)

# Danh sách hình

[Hình 2.1 a) Lược đồ hình sao b) Lược đồ bông tuyết [7] 8](#_Toc130746913)

[Hình 2.2 Các đặc trưng chính của dữ liệu lớn [12] 10](#_Toc130746914)

[Hình 2.3 Ví dụ về DM [16] 14](#_Toc130746915)

[Hình 2.4 Mô hình dữ liệu đồ thị cơ bản 16](#_Toc130746916)

[Hình 2.5 Kiến trúc xây dựng NoSQL graph data warehouse được đề ra của nhóm tác giả [3] 19](#_Toc130746917)

[Hình 2.6 Kiến trúc của Graph BI và khuôn khổ phân tích được đề xuất ở công trình [8]. 20](#_Toc130746918)

[Hình 2.7 Lược đồ quan hệ hình sao ban đầu [4] 22](#_Toc130746919)

[Hình 2.8 Mô hình dữ liệu đồ thị sau khi được chuyển đồ từ mô hình dữ liệu hình sao [4] 23](#_Toc130746920)

[Hình 2.9 Lược đồ bông tuyết ban đầu [4] 24](#_Toc130746921)

[Hình 2.10 Lược đồ đồ thị sau khi chuyển đổi từ lược đồ bông tuyết [4] 25](#_Toc130746922)

[Hình 2.11 Hiệu suất của các lược đồ khi thực hiện ghi dữ liệu [4] 26](#_Toc130746923)

[Hình 2.12 Hai câu truy vấn không phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4] 27](#_Toc130746924)

[Hình 2.13. Hiệu suất của các lược đồ khi thực hiện truy vấn các dữ liệu không phân cấp [4] 27](#_Toc130746925)

[Hình 2.14 Hai câu truy vấn có phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4] 28](#_Toc130746926)

[Hình 2.15 Hiệu suất của các lược đồ khi thực hiện truy vấn các dữ liệu có phân cấp [4] 29](#_Toc130746927)

[Hình 3.1 Kiến trúc đồ thị của nghiên cứu [6] 41](#_Toc130746928)

[Hình 3.2 Kiến trúc đồ thị của nghiên cứu [9] 42](#_Toc130746929)

[Hình 3.3 Kiến trúc đồ thị đề xuất cho bài toán 43](#_Toc130746930)

[Hình 3.4 Ví dụ cụ thể áp dụng kiến trúc đồ thị đề xuất 49](#_Toc130746931)

[Hình 4.1 Kiến trúc tổng quát của hệ thống 53](#_Toc130746932)

[Hình 4.2 Ví dụ minh họa cho việc rút trích thực thể trong văn bản từ mô hình rút trích thực thể 54](#_Toc130746933)

[Hình 4.3 Mô hình thiết kế SDM 57](#_Toc130746934)

[Hình 4.4 Mô hình thiết kế CDM 60](#_Toc130746935)

[Hình 4.5 Mô hình thiết kế UDM 64](#_Toc130746936)

[Hình 4.6 Mô hình thiết kế CCDM 66](#_Toc130746937)

[Hình 5.1 Biểu đồ xu hướng của các hệ quản trị CSDL đồ thị [26] 68](#_Toc130746938)

[Hình 5.2 Áp dụng công nghệ vào kiến trúc của hệ thống 74](#_Toc130746939)

[Hình 6.1 Sơ đồ kiến trúc chi tiết các công nghệ sử dụng xây dựng thử nghiệm hệ thống GDW trên nền tảng dữ liệu lớn 75](#_Toc130746940)

[Hình 6.2 Sơ đồ cách xử lý dữ liệu khóa học và bài đăng tuyển dụng trong Spark 80](#_Toc130746941)

[Hình 6.3 Khung dữ liệu đầu vào 82](#_Toc130746942)

[Hình 6.4 Kết quả được trả về sau khi nhận diện thực thể 83](#_Toc130746943)

[Hình 6.5 Sơ đồ cách xử lý dữ liệu User trong Spark 88](#_Toc130746944)

[Hình 6.6 Quá trình đổ dữ liệu từ GDW vào SDM 90](#_Toc130746945)

[Hình 6.7 Quá trình đổ dữ liệu từ GDW vào CDM 91](#_Toc130746946)

[Hình 6.8 Quá trình đổ dữ liệu từ GDW vào UDM 92](#_Toc130746947)

[Hình 6.9 Quá trình đổ dữ liệu từ GDW vào CCDM 93](#_Toc130746948)

[Hình 6.10 Kiến trúc các đường ống dữ liệu trong hệ thống 94](#_Toc130746949)

[Hình 6.11 Biểu đồ trực quan dữ liệu khóa học trong CDM 98](#_Toc130746950)

[Hình 6.12 Biểu đồ trực quan dữ liệu về nghề nghiệp trong SDM 99](#_Toc130746951)

[Hình 6.13 Biểu đồ trực quan dữ liệu người dùng trong UDM 99](#_Toc130746952)

[Hình 6.14 Biểu đồ trực quan trình tự của các khóa học trong CCDM 100](#_Toc130746953)

[Hình 6.15 Câu truy vấn sử dụng ở SDM 101](#_Toc130746954)

[Hình 6.16 Kết quả truy vấn phục vụ bài toán ở SDM 101](#_Toc130746955)

[Hình 6.17 Kết quả truy vấn với các thông tin về đồ thị ở SDM 102](#_Toc130746956)

[Hình 6.18 Số lượng bài đăng của nghề nghiệp “data engineer” 102](#_Toc130746957)

[Hình 6.19 Số lượng bài đăng yêu cầu kỹ năng “python” cho nghề nghiệp “data engineer” 102](#_Toc130746958)

[Hình 6.20 Câu truy vấn sử dụng ở CDM 103](#_Toc130746959)

[Hình 6.21 Kết quả truy vấn các khóa học có hướng dẫn ngôn ngữ lập trình “python” ở CDM 103](#_Toc130746960)

[Hình 6.22 Kết quả truy vấn với các thông tin về đồ thị CDM 104](#_Toc130746961)

[Hình 6.23 Câu truy vấn được sử dụng ở UDM 104](#_Toc130746962)

[Hình 6.24 Kết quả truy vấn thông tin của một người dùng có username là “toan” ở UDM 105](#_Toc130746963)

[Hình 6.25 Kết quả truy vấn với các thông tin về đồ thị UDM 105](#_Toc130746964)

[Hình 6.26 Câu truy vấn sử dụng ở CCDM 106](#_Toc130746965)

[Hình 6.27 Kết quả truy vấn cho tập hợp các khóa học cần học cho khóa học có id là 178 ở CCDM 106](#_Toc130746966)

[Hình 6.28 Kết quả với các thông tin về đồ thị CCDM 106](#_Toc130746967)

[Hình 6.29 Lược đồ kho dữ liệu quan hệ thử nghiệm 108](#_Toc130746968)

[Hình 6.30 Câu truy vấn không phân cấp Q1 sử dụng SQL 113](#_Toc130746969)

[Hình 6.31 Câu truy vấn không phân cấp Q1 sử dụng Cypher 113](#_Toc130746970)

[Hình 6.32 Biểu đồ trực quan kết quả thời gian truy vấn không phân cấp Q1 của hai hệ thống 114](#_Toc130746971)

[Hình 6.33 Câu truy vấn không phân cấp Q2 sử dụng SQL 115](#_Toc130746972)

[Hình 6.34 Câu truy vấn không phân cấp Q2 sử dụng Cypher 115](#_Toc130746973)

[Hình 6.35 Biểu đồ trực quan kết quả thời gian truy vấn không phân cấp Q2 của hai hệ thống 116](#_Toc130746974)

[Hình 6.36 Câu truy vấn có phân cấp Q3 (mức độ phân cấp 2) sử dụng SQL 116](#_Toc130746975)

[Hình 6.37 Câu truy vấn có phân cấp Q3 (mức độ phân cấp 2) sử dụng Cypher 117](#_Toc130746976)

[Hình 6.38 Biểu đồ trực quan kết quả thời gian truy vấn phân cấp Q3 (mức độ phân cấp 2) của hai hệ thống 118](#_Toc130746977)

[Hình 6.39 Câu truy vấn có phân cấp Q4 (mức độ phân cấp 3) sử dụng SQL 119](#_Toc130746978)

[Hình 6.40 Câu truy vấn có phân cấp Q4 (mức độ phân cấp 2) sử dụng Cypher 119](#_Toc130746979)

[Hình 6.41 Biểu đồ trực quan kết quả thời gian truy vấn phân cấp của hai hệ thống 120](#_Toc130746980)

# Danh sách bảng

[Bảng 2.1 Bảng đối sánh giữa DM và DW [14] 14](#_Toc130746981)

[Bảng 2.2 Bảng so sánh kiến trúc giữa kho dữ liệu đồ thị và kho dữ liệu quan hệ 30](#_Toc130746982)

[Bảng 3.1 Yêu cầu 1 của bài toán 35](#_Toc130746983)

[Bảng 3.2 Yêu cầu 2 của bài toán 36](#_Toc130746984)

[Bảng 3.3 Yêu cầu 3 của bài toán 36](#_Toc130746985)

[Bảng 3.4 Yêu cầu 4 của bài toán 37](#_Toc130746986)

[Bảng 3.5 Yêu cầu 5 của bài toán 38](#_Toc130746987)

[Bảng 3.6 Yêu cầu 6 của bài toán 39](#_Toc130746988)

[Bảng 3.7 Các thực thể trong kiến trúc đồ thị đề xuất 43](#_Toc130746989)

[Bảng 3.8 Các thuộc tính của thực thể trong kiến trúc đồ thị đề xuất 44](#_Toc130746990)

[Bảng 3.9 Các mối quan hệ trong kiến trúc đồ thị đề xuất 46](#_Toc130746991)

[Bảng 4.1 Giải pháp thiết kế DM cho nhóm chức năng phân tích thống kê 55](#_Toc130746992)

[Bảng 4.2 Mô tả thành phần thực thể trong SDM 57](#_Toc130746993)

[Bảng 4.3 Giải pháp thiết kế DM cho nhóm chức năng tìm kiếm 58](#_Toc130746994)

[Bảng 4.4 Mô tả thành phần thực thể trong CDM 60](#_Toc130746995)

[Bảng 4.5 Giải pháp thiết kế DM cho nhóm chức năng tư vấn 62](#_Toc130746996)

[Bảng 4.6 Mô tả thành phần thực thể trong UDM 64](#_Toc130746997)

[Bảng 4.7 Mô tả thành phần thực thể trong CCDM 66](#_Toc130746998)

[Bảng 5.1 Bảng đối sánh giữa các CSDL đồ thị [9] 69](#_Toc130746999)

[Bảng 5.2 So sánh các công nghệ hỗ trợ xử lý dữ liệu lớn [28] [29] [30] [31] [32] 72](#_Toc130747000)

[Bảng 6.1 Danh sách các công nghệ được sử dụng xây dựng hệ thống 75](#_Toc130747001)

[Bảng 6.2 Các tệp lưu trữ các tập tin trong hệ thống lưu trữ 77](#_Toc130747002)

[Bảng 6.3 Mô tả về các cột được lưu trữ trong các dữ liệu thô khóa học, bài đăng tuyển dụng và người dùng 77](#_Toc130747003)

[Bảng 6.4 Bảng chi tiết các trường được trả về từ khung dữ liệu của Spark NLP [33] 81](#_Toc130747004)

[Bảng 6.5 Các mô tả cấu hình trong Spark Neo4j Connector trên hệ thống thử nghiệm [35] 84](#_Toc130747005)

[Bảng 6.6 Cấu hình của máy ảo 95](#_Toc130747006)

[Bảng 6.7 Bảng chi tiết thời gian chạy của các đường ống dữ liệu 95](#_Toc130747007)

[Bảng 6.8 Bảng mô tả chi tiết các bảng và ý nghĩa của chúng trong TDW 109](#_Toc130747008)

[Bảng 6.9 Bảng chi tiết các trường hợp phân cấp và các bảng/nút và quan hệ trong câu truy vấn của hai hệ thống 111](#_Toc130747009)

[Bảng 6.10 Bảng kết quả thời gian truy vấn không phân cấp Q1 113](#_Toc130747010)

[Bảng 6.11 Bảng kết quả thời gian truy vấn không phân cấp Q2 của hai hệ thống với khối lượng dữ liệu khác nhau 115](#_Toc130747011)

[Bảng 6.12 Bảng kết quả thời gian truy vấn phân cấp Q3 (mức độ phân cấp 2) của hai hệ thống với khối lượng dữ liệu khác nhau 117](#_Toc130747012)

[Bảng 6.13 Bảng kết quả thời gian truy vấn có phân cấp Q4 (mức độ phân cấp 3-2) của hai hệ thống với khối lượng dữ liệu khác nhau 119](#_Toc130747013)

[Bảng 6.14 Đánh giá mức độ hoàn thiện các chức năng trong hệ thống 122](#_Toc130747014)

# Danh sách từ viết tắt

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Từ** | **Tên đầy đủ** | **Ý nghĩa** |
|  | BI | Business Intelligence | Kinh doanh thông minh |
|  | CCDM | Course-Course Data Mart | Kho dữ liệu liên kết các khóa học |
|  | CDM | Course Data Mart | Kho dữ liệu khóa học |
|  | CNTT | Công nghệ thông tin | Công nghệ thông tin |
|  | CSDL | Cơ sở dữ liệu | Cơ sở dữ liệu |
|  | DM | Data Mart | Kho dữ liệu theo nhóm chức năng |
|  | DW | Data Warehouse | Kho dữ liệu |
|  | GDBMS | Graph Database  Management System | Hệ quản trị cơ sở dữ liệu đồ thị |
|  | GDM | Graph Data Mart | Kho dữ liệu đồ thị theo nhóm chức năng |
|  | GDW | Graph Data Warehouse | Kho dữ liệu đồ thị |
|  | GVHD | Giáo viên hướng dẫn | Giáo viên hướng dẫn |
|  | MOOCs | Massive Open Online Courses | Các khóa học trực tuyến mở đại trà |
|  | NER | Named-Entity Recognition | Nhận diện thực thể có gán nhãn |
|  | NLP | Natural Language Processing | Xử lý ngôn ngữ tự nhiên |
|  | OLAP | Online Analytical Processing | Xử lý và phân tích dữ liệu trực tuyến |
|  | SDM | Statistical Analysis Data Mart | Kho dữ liệu phân tích thống kê |
|  | TDW | Traditional Data Warehouse | Kho dữ liệu truyền thống |
|  | UDM | User Data Mart | Kho dữ liệu người dùng |

# Tóm tắt

Hiện nay, do sự bùng nổ về kích thước và sự đa dạng của dữ liệu, hay còn gọi với hiện tượng “dữ liệu lớn”, nên kho dữ liệu quan hệ truyền thống không còn đáp ứng được nhu cầu lưu trữ dữ liệu cũng như thực hiện phân tích và xử lý chúng. Do đó, đề tài nghiên cứu nhằm xây dựng một kho dữ liệu đồ thị dựa trên nền tảng dữ liệu lớn, với mục đích nghiên cứu và áp dụng, cung cấp dữ liệu cho ứng dụng thống kê xu hướng nghề nghiệp, kỹ năng nghề nghiệp và tư vấn khóa học cho mục tiêu nghề nghiệp trong lĩnh vực công nghệ thông tin cho các sinh viên.

Đề tài sử dụng các công cụ xử lý dữ liệu lớn và kho dữ liệu đồ thị để xử lý và lưu trữ các thông tin về các bài đăng tuyển dụng, các khóa học được cung cấp bởi các nền tảng cung cấp các khóa học trực tuyến mở đại trà vào kho dữ liệu đồ thị. Sản phẩm đầu ra của đề tài là hệ thống kho dữ liệu đồ thị phục vụ cho hệ thống tư vấn nghề nghiệp và lộ trình học tập cho người học trong phạm vi lĩnh vực công nghệ thông tin.

Kết quả đạt được là hệ thống thành công áp dụng công cụ xử lý dữ liệu lớn và lưu trữ dữ liệu đồ thị phục vụ cho bài toán, đồng thời xây dựng được hệ thống kho dữ liệu truyền thống tương ứng để lập bảng so sánh và chứng minh được phương pháp áp dụng mô hình đồ thị cho kiến trúc của hệ thống là hợp lý.

Hướng phát triển chính của đề tài là tối ưu hóa quá trình xử lý dữ liệu, thực hiện phân tán đồng thời cải tiến mô hình nhận diện thực thể và xây dựng mô hình dự đoán tên nghề nghiệp trên các bài đăng tuyển dụng.

**Chương 1**

# Mở đầu

*Nội dung của chương này giới thiệu chung về đề tài bao gồm đặt vấn đề đưa ra lý do chọn đề tài, tổng quát về nội dung. Tiếp đó đưa ra mục tiêu, hướng tiếp cận và đóng góp của đề tài. Ở cuối chương trình bày bố cục của luận văn.*

## Giới thiệu chung

Trong trí tuệ kinh doanh thông minh (Business Intelligence), kho dữ liệu (Data Warehouse) là một thành phần cốt lõi và đóng vai trò như một nguồn tri thức hỗ trợ đưa ra quyết định và cung cấp một góc nhìn toàn diện về doanh nghiệp [1]. Mục đích chính của kho dữ liệu là giúp cho doanh nghiệp quản lý và phân tích dữ liệu lịch sử, giúp họ có thể định hướng chiến lược và từ đó tối ưu hóa lợi thế cạnh tranh trên thị trường. Từ năm 1990, sau khi Bill Inmon đề xuất ra kho dữ liệu, thì nó đã dần thay thế cho cơ sở dữ liệu vận hành (Operational Database) và cơ sở dữ liệu giao dịch (Transactional Database) trong việc phục vụ phân tích dữ liệu, bởi các tính chất đặc thù bao gồm: tích hợp (integrated), gán nhãn thời gian và có tính lịch sử (time variant), vận hành ổn định (non-volatile) và hướng chủ đề (subject-oriented) [2].

Sau nhiều năm phát triển, kiến trúc và mô hình của kho dữ liệu truyền thống đã trở nên ổn định và vững chắc, chúng được xây dựng theo quy trình chuẩn và áp dụng rộng rãi ở giới công nghiệp. Cũng chính vì điều đó, trong những năm gần đây kho dữ liệu đã không thể đáp ứng được “dữ liệu lớn” (Big Data) - một hiện tượng bùng nổ về kích thước và sự đa dạng của dữ liệu. Do đó, các nhà nghiên cứu, cũng như giới công nghiệp đã dần chuyển sang sử dụng cơ sở dữ liệu NoSQL (Not only SQL) thay cho cơ sở dữ liệu quan hệ nhằm cải thiện những khuyết điểm của kho dữ liệu truyền thống về tính linh hoạt và khả năng mở rộng [3].

Trong bốn loại cơ sở dữ liệu NoSQL được áp dụng vào kho dữ liệu để thay thế kho dữ liệu truyền thống, gồm: cơ sở dữ liệu khóa-giá trị, cơ sở dữ liệu tài liệu, cơ sở dữ liệu dạng cột và cơ sở dữ liệu đồ thị thì áp dụng cơ sở dữ liệu đồ thị vào kho dữ liệu để xây dựng kho dữ liệu đồ thị (Graph-based Data Warehouse) được cho là giải pháp tốt hơn so với các loại NoSQL còn lại. Không chỉ bởi vì tính chất liên kết của các thực thể dữ liệu trên thực tế, mà còn bởi phương pháp tiếp cận cho việc thiết kế mô hình dữ liệu đồ thị dễ dàng hơn so với các loại mô hình NoSQL khác và mô hình dữ liệu quan hệ truyền thống [3] [4].

Dựa trên nhu cầu và giá trị của kho dữ liệu đồ thị được quan tâm bởi các nghiên cứu hiện nay, đề tài luận văn “Xây dựng kho dữ liệu đồ thị dựa trên nền tảng dữ liệu lớn cho đề xuất khóa học dựa trên mục tiêu nghề nghiệp CNTT” tập trung nghiên cứu xây dựng một kho dữ liệu đồ thị phục vụ cho hệ thống tư vấn nghề nghiệp và lộ trình học tập cho người học trong phạm vi lĩnh vực công nghệ thông tin. Trong đó, kho dữ liệu đồ thị được sử dụng như một nguồn tri thức và các kho dữ liệu đồ thị theo nhóm chức năng (Graph Data Mart) được xây dựng để phục vụ trả lời cho các nhu cầu thống kê, phân tích xu hướng của nghề nghiệp, kỹ năng nghề nghiệp cũng như tư vấn các khóa học theo mục tiêu nghề nghiệp trong lĩnh vực Công nghệ thông tin (CNTT)

## Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng một hệ thống kho dữ liệu đồ thị phục vụ cho hệ thống tư vấn nghề nghiệp và lộ trình học tập cho người học trong phạm vi lĩnh vực CNTT. Hệ thống sử dụng dữ liệu trong hệ thống kho dữ liệu đồ thị để tìm ra xu hướng kỹ năng và nghề nghiệp, tìm kiếm chọn lọc khóa học và xây dựng lộ trình các khóa học có trình tự để người dùng có thể tham khảo và bổ sung kiến thức cần thiết cho ngành nghề mình mong muốn.

Về mặt chuyên môn, mục tiêu cần đạt được:

* Tìm hiểu và nắm vững kiến thức về kho dữ liệu, các khái niệm và các loại kho dữ liệu đã có.
* Tìm hiểu và tổng hợp các bài báo nghiên cứu về kho dữ liệu đồ thị trong những năm gần đây, đưa ra được tổng quan về kho dữ liệu đồ thị cho đến hiện tại.
* Tìm hiểu, thiết kế và xây dựng kho dữ liệu đồ thị.
* Tìm hiểu và nắm vững kiến thức về kho dữ liệu theo nhóm chức năng, cách thiết kế và xây dựng chúng.
* Tìm hiểu về hệ thống tư vấn để có thể áp dụng thiết kế được một kho dữ liệu đồ thị phù hợp.
* Tìm hiểu về dữ liệu lớn, áp dụng mô hình máy học và các công cụ hỗ trợ xử lý phù hợp áp dụng cho đề tài.
* Đưa ra được bảng so sánh chi tiết về kho dữ liệu đồ thị và kho dữ liệu quan hệ truyền thống.
* Cài đặt và thử nghiệm hệ thống kho dữ liệu đồ thị cho đề tài.

Về mặt kỹ năng, cần phải đạt được các tiêu chí sau:

* Kỹ năng lập kế hoạch và phân chia thời gian.
* Kỹ năng nghiên cứu và đọc hiểu các tài liệu khoa học.
* Kỹ năng giải quyết vấn đề.
* Kỹ năng tiếp cận công nghệ mới.
* Kỹ năng làm việc nhóm.
* Kỹ năng trình bày báo cáo và thuyết trình.

Về mặt kỹ thuật, cần đạt được:

* Ngôn ngữ lập trình: Python.
* Hệ quản trị cơ sở dữ liệu đồ thị Neo4j và ngôn ngữ truy vấn đồ thị Cypher.
* Công cụ xử lý dữ liệu lớn Apache Spark.

## Hướng tiếp cận đề tài

Để có thể hoàn thành tốt đề tài này, chúng em đưa ra hướng tiếp cận như sau:

* Tổng quan các tài liệu nghiên cứu: Tổng hợp lại các tài liệu nghiên cứu, các nguồn thông tin đã có liên quan tới kho dữ liệu đồ thị NoSQL để phục vụ cho nghiên cứu của đề tài.
* Thiết kế hệ thống cho đề tài: lựa chọn phương pháp nghiên cứu, tìm hiểu để lưu trữ và phân tích kho dữ liệu đồ thị, các công cụ xử lý dữ liệu lớn, đưa ra mô hình dữ liệu đồ thị cho kho dữ liệu đồ thị, kho dữ liệu theo nhóm chức năng và kiến trúc hệ thống cho đề tài hệ thống tư vấn nghề nghiệp và lộ trình học.
* Thử nghiệm và đánh giá hệ thống: dựa trên thiết kế được đưa ra và tiến hành cài đặt hệ thống, sau đó thử nghiệm hệ thống có ưu điểm gì so với kho dữ liệu truyền thống thông qua các tiêu chuẩn: tiêu chuẩn khách quan (tốc độ ghi dữ liệu, tốc độ đọc dựa vào các câu truy vấn, khả năng mở rộng), tiêu chuẩn chủ quan (khả năng hỗ trợ, độ linh hoạt, và dễ cài đặt hay không) [5].

Mô hình xây dựng: Dựa trên các mô hình kiến trúc đồ thị đã được thiết kế và dữ liệu về khóa học và nghề nghiệp được cung cấp từ kho lưu trữ dữ liệu thô, tiến hành cài đặt quy trình triển khai xây dựng kho dữ liệu đồ thị dựa trên nền tảng dữ liệu lớn cho đề xuất khóa học dựa trên mục tiêu nghề nghiệp CNTT bao gồm các bước sau:

* Triển khai xây dựng kho dữ liệu đồ thị:
  + Tập hợp và lưu trữ các tập tin dữ liệu vào kho dữ liệu thô
  + Lấy dữ liệu từ kho lưu trữ dữ liệu thô.
  + Sử dụng các công cụ xử lý dữ liệu lớn để triển khai mô hình nhận diện thực thể, nguồn do GVHD cung cấp từ đề tài [6] kết hợp tiền xử lý dữ liệu trước khi đổ vào kho.
  + Thực hiện đổ dữ liệu đã được xử lý vào kho dữ liệu đồ thị theo mô hình đã được thiết kế.
* Triển khai xây dựng các kho dữ liệu theo nhóm chức năng:
  + Lấy dữ liệu từ kho dữ liệu đồ thị.
  + Thực hiện xử lý dữ liệu và đổ dữ liệu vào mô hình đã được thiết kế theo từng yêu cầu của bài toán.

Các phương pháp nghiên cứu khoa học được áp dụng trong đề tài luận văn: Phương pháp phân tích - tổng hợp, phương pháp so sánh, phương pháp liệt kê, phương pháp thu thập số liệu, phương pháp thực nghiệm, ….

## Đóng góp của đề tài

Đề tài xây dựng một kho dữ liệu đồ thị dựa trên nền tảng dữ liệu lớn phục vụ cho hệ thống tư vấn nghề nghiệp và lộ trình học tập cho người học trong lĩnh vực CNTT. Hệ thống tư vấn sẽ sử dụng dữ liệu trong hệ thống kho dữ liệu đồ thị để tìm ra xu hướng kỹ năng và nghề nghiệp, tìm kiếm chọn lọc khóa học và xây dựng lộ trình các khóa học có trình tự để người dùng có thể tham khảo về nghề nghiệp, kỹ năng và bổ sung kiến thức cần thiết cho ngành nghề mình mong muốn.

## Bố cục trình bày

Cuốn báo cáo luận văn gồm 7 chương, nội dung tóm tắt của các chương được liệt kê như sau:

* **Chương 1 - Mở đầu:** Trình bày vấn đề tổng quan của kho dữ liệu truyền thống, đặt vấn đề (lý do chọn đề tài), giới thiệu chung đề tài (về kho dữ liệu đồ thị cho hệ thống tư vấn thời gian thực trên nền tảng dữ liệu lớn), phạm vi đề tài, mục tiêu đề tài, hướng tiếp cận, đóng góp và bố cục của đề tài.
* **Chương 2 - Tổng quan về kho dữ liệu và kho dữ liệu đồ thị:** Các kiến thức cơ bản về kho dữ liệu, tình trạng của kho dữ liệu, kho dữ liệu theo nhóm chức năng. Tổng quan về kho dữ liệu đồ thị và các bài công trình nghiên cứu kho dữ liệu đồ thị. So sánh kho dữ liệu quan hệ truyền thống và kho dữ liệu đồ thị.
* **Chương 3 - Kiến trúc dữ liệu đồ thị đề xuất:** Phát biểu phạm vi bài toán, mô tả các yêu cầu chức năng, cơ sở xác định lớp thực thể, thiết kế kiến trúc đồ thị và mô tả, cuối cùng phân tích cơ sở khoa học về mô hình đồ thị.
* **Chương 4 - Thiết kế kiến trúc tổng quát của hệ thống:** Thiết kế kiến trúc logic hệ thống, thiết kế và mô tả các kho dữ liệu theo nhóm chức năng.
* **Chương 5 - Phân tích lựa chọn công nghệ triển khai hệ thống:** Mô tả nhiệm vụ và nhóm công nghệ tương ứng trong kiến trúc của hệ thống, phân tích và lựa chọn công nghệ phù hợp với bài toán của đề tài.
* **Chương 6 - Thiết kế triển khai và cài đặt thử nghiệm:** Mô tả kiến trúc triển khai cụ thể, công nghệ sử dụng, mô tả chi tiết dữ liệu chạy trên hệ thống, mô tả phương pháp thử nghiệm và quy trình chạy thử nghiệm và kết quả. Cuối cùng phân tích và đánh giá nhận xét kết quả đạt được.
* **Chương 7 - Kết luận và hướng phát triển:** Trình bày kết quả và kinh nghiệm đạt được sau khi thực hiện đề tài. Nêu hướng phát triển của đề tài và những đề xuất cải tiến hệ thống xây dựng kho dữ liệu đồ thị cho mục tiêu bài toán trong tương lai.

**Chương 2**

# Tổng quan về kho dữ liệu và kho dữ liệu đồ thị

*Nội dung của chương này trình bày sơ lược, hiện trạng của kho dữ liệu trong những năm gần đây, đồng thời giới thiệu, tổng quan về kho dữ liệu theo nhóm chức năng, kho dữ liệu đồ thị và các bài công trình nghiên cứu kho dữ liệu đồ thị. Ở cuối chương thực hiện so sánh kho dữ liệu quan hệ và kho dữ liệu đồ thị, cuối cùng đưa ra lý do sử dụng kho dữ liệu đồ thị áp dụng vào đề tài khóa luận.*

## Kho dữ liệu và tình trạng của kho dữ liệu hiện nay

### Kho dữ liệu

Kho dữ liệu (Data warehouse, viết tắt là DW) là cơ sở dữ liệu (CSDL) lịch sử, được tạo ra để nhằm mục đích hỗ trợ việc đưa ra quyết định cho giới công nghiệp. DW lấy dữ liệu từ các CSDL giao dịch/vận hành và các nguồn dữ liệu khác, sau đó biến chúng thành một cấu trúc thống nhất với mô hình dữ liệu được thiết kế sẵn để phù hợp cho việc thực hiện phân tích kinh doanh và đưa ra quyết định [2]. Để đáp ứng được nhu cầu trên, DW có bốn tính chất đặc thù:

* Tích hợp (integrated)
* Gán nhãn thời gian và có tính lịch sử (time variant)
* Vận hành ổn định (non-volatile)
* Hướng chủ đề (subject-oriented)

Trong DW, dữ liệu có thể được xây dựng theo các lược đồ khác nhau, chủ yếu là lược đồ hình sao và lược đồ bông tuyết. Điểm chung của 2 lược đồ là đều bao gồm bảng sự kiện (fact) gồm các phép đo (measure), số liệu (metric) về một quy trình kinh doanh và các bảng chiều (dimension) là các thực thể chứa các thuộc tính trên bảng sự kiện để tính toán số liệu.

Shape

Description automatically generated

Hình . a) Lược đồ hình sao b) Lược đồ bông tuyết [7]

Trên hình 2.1a, lược đồ hình sao gồm một bảng sự kiện trung tâm và xung quanh gồm các bảng chiều tạo thành một hình ngôi sao, bảng sự kiện sẽ được liên kết với các bảng chiều thông qua khóa ngoại để cung cấp thông tin chi tiết. Lược đồ này được tối ưu hóa cho việc truy vấn bộ dữ liệu lớn do có các bảng chiều không được chuẩn hóa nên số lượng bảng cần ghép trong câu truy vấn không quá nhiều. Ở hình 2.1b, lược đồ bông tuyết giống với lược đồ hình sao, cũng có một bảng sự kiện ở giữa, nhưng các bảng chiều xung quanh được chuẩn hóa, tạo thành các phân cấp thể hiện độ chi tiết cho các khía cạnh thông tin muốn phân tích, đây còn được gọi là độ mịn (granularity) của dữ liệu. Tùy theo nhu cầu phân tích của doanh nghiệp, họ sẽ xây dựng hệ thống DW theo lược đồ hình sao hay lược đồ bông tuyết, dựa vào ưu và nhược điểm của hai loại lược đồ.

Từ những ngày đầu, DW đã được hướng đến với nhiệm vụ là xử lý và phân tích dữ liệu trực tuyến (Online Analytical Processing, viết tắt là OLAP), hệ thống OLAP cho phép người dùng truy vấn, tương tác và tổng hợp dữ liệu có trong DW, do đó có thể dễ dàng truy cập các dữ liệu thông tin cần thiết và phân tích dữ liệu ở các khía cạnh khác nhau.

### Tình trạng của kho dữ liệu hiện nay

Vào những năm 1990, khái niệm DW đã được đề xuất ra bởi Bill Inmon, để giúp cho giới công nghiệp có thể quản lý và lưu trữ dữ liệu lịch sử, từ đó khai thác và phân tích dữ liệu, đưa ra hướng đi và chiếm lợi thế kinh tế trên thị trường. Các CSDL giao dịch thông thường trước đó không được tối ưu cho nhu cầu phân tích và khai thác dữ liệu do mục tiêu ban đầu được thiết kế để phục vụ cho các giao dịch hằng ngày và đảm bảo tính toàn vẹn của dữ liệu. Kể từ đó, DW được lựa chọn như là một giải pháp để có thể lưu trữ, phân tích và khai thác dữ liệu từ đó đưa ra quyết định cho doanh nghiệp [8]. Sau nhiều năm phát triển, kiến trúc và mô hình của DW đã trở nên ổn định và vững chắc, chúng được xây dựng theo quy trình chuẩn và áp dụng rộng rãi ở giới công nghiệp [2].

Đặc biệt, trong những năm gần đây, lượng dữ liệu ngày càng lớn, các nguồn dữ liệu trở nên đa dạng và dữ liệu thay đổi linh động, khiến cho việc quản lý và lưu trữ bằng những công cụ và phương pháp quản lý dữ liệu truyền thống ngày càng khó khăn hơn. Với sự phát triển mạnh mẽ từ mạng xã hội, các thiết bị IoT (Internet of Things), dữ liệu có tính di động và các nguồn khác,... đã khiến cho dữ liệu ngày càng đa dạng, khổng lồ và phức tạp để có thể xử lý và khai thác. Cụm từ “Big Data” - dữ liệu lớn đã được đề cập thường xuyên như một hiện tượng cho những trường hợp trên [2].

Dữ liệu lớn có thể hiểu theo nghĩa đen của nó là một tập dữ liệu khổng lồ, nhưng không chỉ thế, dữ liệu còn có thể tăng dần theo thời gian. Do kích thước quá lớn và phức tạp nên những công cụ xử lý, phân tích và lưu trữ dữ liệu truyền thống không còn hiệu quả. Trong dữ liệu lớn gồm 3 loại dữ liệu: dữ liệu không cấu trúc (unstructured data), dữ liệu bán cấu trúc (semi-structured data) và dữ liệu có cấu trúc (structured data). Dữ liệu lớn có những đặc trưng chính như sau [9] [10] [11]:

Diagram

Description automatically generated

Hình . Các đặc trưng chính của dữ liệu lớn [12]

* Khối lượng dữ liệu (Volume): Đúng với tên gọi của “dữ liệu lớn” bởi bản thân nó đã một kích thước dữ liệu khổng lồ. Một tập dữ liệu có được gọi là “dữ liệu lớn” hay không là phụ thuộc vào độ lớn của dữ liệu.
* Đa dạng dữ liệu (Variety): Dữ liệu hiện nay được tạo ra từ rất nhiều nguồn khác nhau, không chỉ về các dữ liệu có cấu trúc truyền thống, mà gần đây xu hướng dữ liệu là ở dạng hình ảnh, video, âm thanh và nhiều dạng dữ liệu khác, dẫn đến 80% dữ liệu là không có cấu trúc.
* Giá trị dữ liệu (Value): Giá trị của dữ liệu cũng là một yếu tố chính mà người sử dụng dữ liệu sẽ quan tâm đến. Không chỉ về số lượng dữ liệu lưu trữ và xử lý, mà nó còn là lượng dữ liệu có giá trị, có độ tin cậy cao để được lưu trữ, xử lý và phân tích để tìm ra các thông tin có ích cho người dùng.
* Tốc độ dữ liệu (Velocity): Dữ liệu được tạo ra liên tục và biến đổi rất nhanh. Thậm chí, dữ liệu được thu thập, xử lý, phân tích theo thời gian thực.
* Khả năng trực quan (Visualization): Đây là một phần khó trong dữ liệu lớn. Điều này làm cho tất cả lượng dữ liệu khổng lồ trở nên dễ đọc và dễ hiểu.
* Tính xác thực (Veracity): Về các sai lệch, nhiễu và bất thường trong dữ liệu lớn.
* Tính hợp lệ (Validity): Về tính chính xác của dữ liệu cho mục đích sử dụng của nó.
* Tính không ổn định (Volatility): Về khoảng thời gian một lượng lớn dữ liệu được lưu giữ trong CSDL, mất dữ liệu và đột ngột thao tác dữ liệu với các ký tự khác nhau.
* Sự thay đổi (Variability): Liên quan đến sự không nhất quán của dữ liệu, bên cạnh đó là vô số chiều dữ liệu do nhiều loại và nguồn dữ liệu khác nhau.

Với các khái niệm được trình bày ở trên, có thể hiểu được bản chất của dữ liệu lớn và làm thế nào để xác định chúng. Hiện nay, trong thời đại của “dữ liệu lớn”, với số lượng dữ liệu ngày càng tăng cao, dữ liệu ngày càng đa dạng và mối quan hệ giữa các thực thể ngày càng phức tạp, việc sử dụng DW truyền thống đã khó có thể đáp ứng được nhu cầu lưu trữ và phân tích dữ liệu. CSDL truyền thống dù được gọi là CSDL quan hệ, nhưng chúng không thật sự xử lý được các mối quan hệ phức tạp với các dữ liệu thực tế có mối liên kết phức tạp, đặc biệt là dữ liệu ngày nay càng ngày càng nhiều mối quan hệ N:N (nhiều - đến - nhiều), mô hình dữ liệu sẽ ngày càng phức tạp và khó hiểu khi sử dụng mô hình dữ liệu quan hệ để thể hiện. Do CSDL quan hệ được xây dựng dựa trên lý thuyết tập hợp nên chúng sẽ phù hợp cho các loại dữ liệu dạng bảng hoặc tập hợp các dữ liệu. Khuyết điểm khác của CSDL quan hệ là lược đồ dữ liệu phải được xác định trước, các loại dữ liệu của các thuộc tính phải được xác định rõ ràng khi cài đặt và rất khó để thay đổi lược đồ, không phù hợp với dữ liệu thường xuyên thay đổi [13].

Với CSDL thông thường, mỗi lần truy vấn trên nhiều bảng đều cần phải thực hiện phép kết bảng (join). Khi kết bảng, hệ quản trị CSDL phải tìm khóa ngoại để phù hợp với điều kiện kết bảng, và để tìm kiếm khóa ngoại hệ thống sẽ sử dụng chỉ mục (index). Chỉ mục của hệ quản trị CSDL quan hệ sử dụng cấu trúc dữ liệu cây nhị phân (B-tree) để xây dựng, với độ phức tạp của thuật toán là O(log(n)) đã giúp tăng đáng kể tốc độ truy vấn của hệ thống và khi dữ liệu tăng lên 10 lần thì tốc độ truy vấn sẽ giảm đi 2 lần. Nhưng trong tình trạng dữ liệu quá lớn trong những năm gần đây, càng nhiều dữ liệu thì hệ thống CSDL sẽ phải thực hiện tìm kiếm khóa ngoại càng nhiều khi kết bảng, dẫn đến tốc độ thực thi ngày càng chậm [13].

Để có thể quản lý và xử lý được khối lượng dữ liệu như thế, cần phải có giải pháp mới thay thế cho phương pháp và công cụ xử lý truyền thống. Các giải pháp như lưu trữ và xử lý phân tán, CSDL NoSQL và CSDL trên bộ nhớ (In-memory database) được đề ra bởi các nghiên cứu khác nhau. Ngoài ra, các quy trình và công cụ Trích xuất – Biến đổi – Tải (Extract – Transform – Load, thường được viết tắt là ETL) hay Trích xuất – Tải – Biến đổi  (Extract – Load - Transform, thường được viết tắt là ELT) khác nhau cũng được đưa ra để xử lý khối lượng dữ liệu lớn và đa dạng [2].

Đối với các mô hình CSDL NoSQL bao gồm CSDL khóa - giá trị (key-value), CSDL dạng cột (column family) và CSDL dạng tài liệu (document) khi áp dụng vào hệ thống DW đều đem đến các hướng giải quyết khác nhau cho vấn đề trên. Với CSDL khóa - giá trị, mô hình dữ liệu đã giải quyết vấn đề mối quan hệ bằng cách loại bỏ hoàn toàn mối quan hệ khỏi mô hình, tất cả các dữ liệu liên quan đều có sẵn ở dạng khóa - giá trị. Khác với CSDL khóa - giá trị, CSDL dạng cột tuy có thể bỏ qua mối quan hệ trong mô hình dữ liệu với các dữ liệu không có chuẩn hóa, nhưng dẫn đến dữ liệu bị lặp rất nhiều và doanh nghiệp sẽ cần phải mua một lượng lớn phần cứng để xây dựng thêm các máy chủ để mở rộng dung lượng lưu trữ cho CSDL. CSDL dạng tài liệu tốt hơn so với CSDL dạng cột bởi tất cả dữ liệu của một thực thể sẽ được thêm vào một khối liền kề (adjacent block), nó sẽ hoạt động khá tốt khi người dùng không cần tới mối quan hệ giữa các thực thể và sự liên kết giữa chúng. Song, các mô hình dữ liệu NoSQL kể trên vẫn không thể đáp ứng được nhu cầu thể hiện mối quan hệ giữa các thực thể [13].

Do đó, các nghiên cứu về kho dữ liệu đồ thị được nghiên cứu và xem xét như một giải pháp cho các vấn đề được kể trên bởi khi dữ liệu được tổ chức dưới dạng đồ thị, mối quan hệ giữa các thực thể sẽ được thể hiện rõ ràng và các thuật toán duyệt đồ thị sẽ giúp việc truy vấn thông tin hiệu quả hơn. Ở mục 2.3.1 và mục 2.3.2 sẽ giới thiệu sơ lược về đặc tính và điểm mạnh về mô hình CSDL đồ thị để áp dụng vào hệ thống DW. Trong đề tài này, chúng em tập trung nghiên cứu DW đồ thị, sử dụng các yếu tố khác nhau so sánh lợi thế với DW đồ thị quan hệ truyền thống.

## Kho dữ liệu theo nhóm chức năng

### Giới thiệu kho dữ liệu theo nhóm chức năng

Data mart (viết tắt là DM), được biết nhưkho dữ liệu theo nhóm chức năng, nó được định nghĩa như là một phiên bản đơn giản hóa của DW tùy vào nhu cầu phân tích của bài toán, DM sẽ được xây dựng để phục vụ cho các yêu cầu bài toán đó. Một DM sẽ chứa một phần của dữ liệu trong DW gốc, các giới công nghiệp có thể chia ra các bộ phận trong DW thành các DM phục vụ cho các bộ phận nhỏ như nhân sự, tài chính, dây chuyền cung ứng và nhiều nhu cầu khác nữa. Trong hiện tượng “dữ liệu lớn” và nhu cầu phân tích dữ liệu tăng cao, DM giúp doanh nghiệp dễ dàng chuyển đổi các thông tin dữ liệu thành các khía cạnh để phân tích. Không chỉ thế, DM cũng giúp cho doanh nghiệp tăng tốc quy trình nghiệp vụ bởi họ có thể truy cập vào các dữ liệu liên quan trong một DW với khối lượng dữ liệu khổng lồ trong một thời gian ngắn [14] [15].

Diagram

Description automatically generated

Hình . Ví dụ về DM [16]

Sự khác biệt giữa DW và DM chủ yếu ở phần mục tiêu về dữ liệu được lưu trữ: DW được xây dựng như một nơi tập trung dữ liệu của toàn bộ doanh nghiệp, trong khi DM phục vụ chủ yếu cho một bộ phận hay một chức năng nghiệp vụ. Thêm vào đó, truy vấn thông tin cần thiết trong DW lớn là một công việc khó khăn cho doanh nghiệp, do đó, DM được xây dựng để chia nhỏ dữ liệu trong DW thành các nhu cầu bài toán và bộ phận khác nhau để có thể dễ dàng quản lý và phân tích dữ liệu trên đó. Ở hình 2.3 là ví dụ cho việc một DW chứa các dữ liệu về nhân sự, tài chính và chuỗi cung ứng, DW này sau đó được chia nhỏ các chức năng ra thành các kho dữ liệu con tương ứng với từng phòng ban. Sự đối sáng giữa DM và DW được trình bày rõ trong bảng 2.1.

Bảng . Bảng đối sánh giữa DM và DW [14]

|  |  |  |
| --- | --- | --- |
|  | **Kho dữ liệu theo nhóm chức năng (Data mart)** | **Kho dữ liệu (Data warehouse)** |
| **Dung lượng lưu trữ** | Nhỏ. | Lớn. |
| **Chủ đề** | Một chủ đề (về bộ phận hoặc về bài toán nào đó). | Nhiều chủ đề (Toàn bộ doanh nghiệp). |
| **Mục tiêu** | Đáp ứng cho một nghiệp vụ nào đó. | Phục vụ cho cả doanh nghiệp. |
| **Nguồn dữ liệu** | Một hoặc vài nguồn. | Nhiều nguồn khác nhau. |
| **Dữ liệu tích hợp** | Một chủ đề nào đó. | Toàn bộ dữ liệu của doanh nghiệp. |

## Kho dữ liệu đồ thị và ứng dụng

### Mô hình dữ liệu đồ thị

Cấu trúc cơ bản của một mô hình dữ liệu đồ thị bao gồm nút (nodes), cạnh liên kết (relationships), thuộc tính (properties) và nhãn (labels). Nút được dùng để biểu diễn các thực thể, có thể là vật thể như người, sản phẩm hoặc là các biểu thức, phép toán,... Các nút thường được gán một hay nhiều các thuộc tính có dạng khóa-giá trị (key-value) để thể hiện đặc tính của thực thể, ngoài ra dùng để nắm bắt một số siêu dữ liệu (metadata) như phiên bản (version) hoặc thời gian (timestamps). Có thể biểu diễn vai trò của nút trong mô hình dữ liệu, mỗi nút đang biểu diễn thực thể nào bằng cách gán cho chúng một hoặc nhiều nhãn. Và cuối cùng, sử dụng cạnh liên kết để thể hiện mối quan hệ giữa các thực thể, từ đó hình thành được cấu trúc của đồ thị. Mỗi cạnh liên kết thường có loại và hướng giúp hiểu được cách các nút kết nối với xung quanh. Việc kết nối được thực hiện đối với từng thực thể riêng lẻ nên không tồn tại ràng buộc về sự giống nhau trong cách kết nối giữa các nút, vì vậy đồ thị là một giải pháp tuyệt vời để mô hình hóa các miền dữ liệu có cấu trúc thay đổi, có kết nối giữa các thực thể đa dạng và linh động. Với cạnh liên kết, ngoài tên của liên kết, ta cũng có thể gán thuộc tính tương tự như nút để biểu thị trọng số, đặc điểm của mối quan hệ giữa các nút và thông tin về siêu dữ liệu. Hình 2.4, ví dụ về một mô hình dữ liệu đồ thị cơ bản biểu diễn kĩ năng của người dùng. Trong ví dụ, gồm thực thể người dùng (user) và kĩ năng (skill) được biểu diễn bằng các nút, nối với nhau bằng cạnh “HAVE\_SKILL” thể hiện người dùng có kĩ năng được nối, bên cạnh đó nút và cạnh có các cặp khóa-giá trị thể hiện thuộc tính.

Diagram

Description automatically generated

Hình . Mô hình dữ liệu đồ thị cơ bản

Với cấu trúc dữ liệu đồ thị, CSDL đồ thị được thiết kế để lưu trữ dữ liệu mà không cần một mô hình cố định từ trước. Một kho dữ liệu đồ thị (Graph Data Warehouse, viết tắt là GDW) là CSDL đồ thị được thiết kế cho việc tối ưu hóa khả năng xử lý và phân tích dữ liệu. Trong GDW, việc thực hiện các câu truy vấn cũng trở nên dễ dàng hơn khi mối liên kết giữa các nút được lưu trữ sẵn ở ổ đĩa, được đặt tên và có hướng mũi tên sẵn giúp cho tốc độ duyệt đồ thị nhanh hơn. Bên cạnh đó, sau khi tải dữ liệu, các thuật toán lý thuyết đồ thị giúp dễ dàng xử lý các truy vấn ngữ nghĩa bằng cách tính toán đường đi ngắn nhất giữa các nút. GDW chỉ định các kết nối ngay lúc dữ liệu được thêm vào để tránh vấn đề về hiệu suất tìm kiếm chỉ mục cho phép toán ghép bảng khi duyệt đồ thị. Điều này khiến đồ thị trở nên tối ưu khi mô hình có nhiều mối quan hệ chồng chéo.

### Sơ lược về CSDL đồ thị

CSDL đồ thị được xây dựng dựa trên lý thuyết đồ thị, bao gồm các nút (node) thể hiện cho các thực thể và các cạnh (edge) thể hiện cho các mối quan hệ giữa các nút. Lý thuyết đồ thị đã tồn tại được khoảng 300 năm, được cho là thể hiện cấu trúc các thực thể liên kết với nhau giữa một nhóm trong xã hội [7]. Song, trong những năm gần đây, do sự phát triển của mạng xã hội và các bài toán về dữ liệu hóa học, sinh học, web,… và các nguồn dữ liệu đa dạng, các mối quan hệ giữa các thực thể phức tạp khiến cho chi phí phép kết của CSDL quan hệ quá cao, đã thúc đẩy doanh nghiệp xem xét và lựa chọn CSDL đồ thị NoSQL làm nơi lưu trữ dữ liệu và phân tích dữ liệu đồ thị. Các tập đoàn lớn như Linkedin, Facebook và Twitter đã sử dụng CSDL đồ thị lưu trữ thông tin truyền thông xã hội để dễ dàng quản lý và khai thác dữ liệu trên đó [17] [18] [19].

CSDL đồ thị cung cấp các thuật toán để khai thác dữ liệu dạng đồ thị, giúp giới công nghiệp có thể dễ dàng khai thác, phân tích các khía cạnh ẩn trong dữ liệu, như khai phá ra các mối quan hệ ẩn giữa các thực thể trong đồ thị. Nhờ vào các ưu điểm trên, CSDL đồ thị được ứng dụng cho các hệ thống phát hiện gian lận, hệ thống tư vấn,… cũng như áp dụng vào các hệ thống CSDL giao dịch trực tuyến nhờ vào tính “ACID” như các hệ thống CSDL quan hệ. Lợi thế khác của CSDL đồ thị là tính mở rộng và linh hoạt của mô hình, cũng như truy vấn dữ liệu bằng cách duyệt các nút trong đồ thị thay vì phải tốn chi phí kết bảng giữa các bảng ở CSDL quan hệ [4].

Qua các tính chất được kể trên, ta đã có thể thấy được các lợi thế của CSDL đồ thị so với với CSDL quan hệ, nhưng nó vẫn có mặt bất lợi nếu khối lượng dữ liệu trở nên quá lớn, lúc đó cần khung xử lý đồ thị (Graph processing framework) để xử lý khối lượng dữ liệu lớn đó trong môi trường phân tán [2].

### Giới thiệu kho dữ liệu đồ thị và ứng dụng

Trong thập kỷ vừa qua, nhiều nhà nghiên cứu đã nhìn ra được lợi thế của cấu trúc dữ liệu đồ thị, họ đã tiến hành nghiên cứu, áp dụng cấu trúc đồ thị vào kho dữ liệu cho ứng dụng phân tích và khai thác dữ liệu. Từ đó sử dụng để thay thế cho kho dữ liệu quan hệ truyền thống do kho dữ liệu truyền thống chỉ có thể lưu trữ được dữ liệu có cấu trúc, khả năng mở rộng kém và không linh hoạt, tốn chi phí cho cho phép toán ghép bảng giữa các bảng sự kiện và bảng chiều – điều này thể hiện rõ rệt hơn khi là lược đồ bông tuyết. Đồng thời, sự tăng trưởng của khối lượng dữ liệu và mối liên hệ phức tạp giữa các thực thể khiến cho kho dữ liệu quan hệ không còn đáp ứng được các vấn đề đã kể trên [4].

Nhiều nghiên cứu được đề ra về ứng dụng kho dữ liệu đồ thị cho việc phân tích dữ liệu và đưa ra các phương pháp tiếp cận cũng như chuyển đổi từ lược đồ quan hệ truyền thống sang mô hình dữ liệu đồ thị, sau đó đưa ra các yếu tố và khía cạnh khác nhau để so sánh như khả năng lưu trữ, thời gian truy vấn, khả năng mở rộng và khả năng tiếp cận,… để thấy được các lợi thế khác nhau của kho dữ liệu đồ thị so với kho dữ liệu quan hệ truyền thống [4].

### Khảo sát hiện trạng kho dữ liệu đồ thị

Cho đến hiện nay, đã có nhiều công trình nghiên cứu về kho dữ liệu đồ thị trong hệ thống thông tin kinh doanh thông minh. Trong đó, các công trình tiêu biểu có thể cho thấy hiện trạng nghiên cứu về kho dữ liệu đồ thị được khắc họa rõ nét hiện nay là các công trình [3] [8] [4].

Ở công trình nghiên cứu [3], nhóm tác giả đã đưa ra mục tiêu hướng về việc sử dụng các CSDL NoSQL, đặc biệt là CSDL đồ thị để xây dựng kho dữ liệu thay cho CSDL quan hệ truyền thống do sự tăng trưởng nhanh chóng của hiện tượng dữ liệu lớn, tác giả đi vào sâu tính chất và ứng dụng của CSDL đồ thị áp dụng cho mô hình dữ liệu hiện nay. Nhóm tác giả cũng nêu ra tổng quan nghiên cứu từ các nghiên cứu trước đó về kho dữ liệu NoSQL dạng cột và kho dữ liệu NoSQL dạng tài liệu, sau đó tổng hợp lại cách tiếp cận về hai loại kho dữ liệu này. Trong bài nghiên cứu, nhóm tác giả đặt trọng tâm về kho dữ liệu NoSQL dạng đồ thị, đưa ra những cơ hội và thách thức khi áp dụng kho dữ liệu đồ thị trong thời đại tăng trưởng nhanh chóng của mạng xã hội và ứng dụng vào các hệ thống tư vấn và đưa ra quyết định. Ở phần cuối của bài nghiên cứu, tác giả đã đề xuất ra một cách tiếp cận mới về kho dữ liệu đồ thị, áp dụng cho dữ liệu xã hội để lưu trữ và phân tích.

Diagram

Description automatically generated

Hình . Kiến trúc xây dựng NoSQL graph data warehouse được đề ra của nhóm tác giả [3]

Trong hình 2.5 thể hiện kiến trúc được đưa ra sử dụng cơ dữ dữ liệu NoSQL dạng cột để lưu trữ thông tin được chia sẻ ở các trang mạng xã hội thay vì sử dụng CSDL quan hệ, sau đó thực hiện quy trình ETL đưa dữ liệu về kho dữ liệu đồ thị để lưu trữ và sử dụng các thuật toán đồ thị để khai thác dữ liệu và áp dụng phân tích xử lý trực tuyến cho đồ thị.

Ở công trình nghiên cứu [8], nhóm tác giả đã cung cấp cái nhìn tổng quan về ứng dụng, quy trình và đề xuất kiến trúc của kho dữ liệu đồ thị vào hệ thống thông tin phục vụ trí tuệ kinh doanh, hay còn được gọi là hệ thống trí tuệ kinh doanh thông minh đồ thị (Graph BI). Đồng thời, nhóm tác giả cũng đưa ra cơ hội, thách thức cũng như hướng đi tương lai của việc ứng dụng kho dữ liệu đồ thị vào hệ thống trí tuệ kinh doanh và sử dụng các thuật toán khai thác đồ thị vào để phục vụ cho việc đưa ra quyết định cũng như xây dựng hệ thống tư vấn. Thông qua công trình nghiên cứu này, người đọc sẽ có được cái nhìn tổng quan về hệ thống Graph BI cũng như tính ứng dụng của nó trong thời đại tăng trưởng của dữ liệu lớn.

Diagram

Description automatically generated

Hình . Kiến trúc của Graph BI và khuôn khổ phân tích được đề xuất ở công trình [8].

Trong hình 2.6, kiến trúc của hệ thống đồ thị BI được chia làm 3 pha:

1. Trích xuất và xây dựng đồ thị
2. Thiết kế không gian đa chiều và tính toán trên khối đồ thị
3. Phân tích trên khối đồ thị

Theo kiến trúc ở hình 2.6, ở pha đầu tiên - trích xuất và xây dựng đồ thị, tác giả mô tả tổng quan quy trình ETL hệ thống kho dữ liệu đồ thị. Các nguồn dữ liệu sẽ được trích xuất, sau đó tiến hành xây dựng cấu trúc đồ thị và xử lý dữ liệu theo cấu trúc đó. Trong quá trình xử lý dữ liệu, có thể sử dụng thêm các mô hình học máy như truy hồi thông tin và xử lý ngôn ngữ tự nhiên để xây dựng đồ thị đa chiều từ các dữ liệu phi cấu trúc như các dữ liệu dạng văn bản. Cuối cùng, dữ liệu sau khi xử lý sẽ được lưu trữ trong kho dữ liệu đồ thị để làm nguồn tri thức cho các công việc phân tích dữ liệu ở các pha sau đó.

Tiếp đến pha thứ 2 - thiết kế không gian đa chiều và tính toán trên khối đồ thị, ở pha này chủ yếu xây dựng các khối đồ thị và mô hình đa chiều dựa trên nguồn dữ liệu từ kho dữ liệu đồ thị được xây dựng lên trước đó tùy theo các nhu cầu phân tích khác nhau. Ở pha cuối - phân tích trên khối đồ thị, người dùng cuối của hệ thống BI có thể sử dụng nguồn tri thức được xây dựng ở 2 pha trước đó để tiến hành thực hiện việc phân tích dữ liệu trên đó. Khác với hệ thống OLAP quan hệ truyền thống, khối đồ thị cũng cho phép người dùng sử dụng các thuật toán đồ thị để khai thác các khía cạnh như xác định cộng đồng hay tìm ra các mối liên kết ẩn giữa các nút để phục vụ cho việc phân tích dữ liệu mạng xã hội dễ dàng hơn.

Ở công trình [4] được công bố gần nhất vào tháng 5/2022, các tác giả đặt trọng tâm về việc so sánh hiệu suất của các lược đồ dữ liệu của kho dữ liệu quan hệ áp dụng vào mô hình dữ liệu đồ thị. Đồng thời, nhóm tác giả cũng đưa ra các quy luật chuyển đổi từ lược đồ hình sao sang lược đồ hình sao dạng đồ thị, từ lược đồ bông tuyết sang lược đồ bông tuyết dạng đồ thị.

Công trình nghiên cứu đưa ra chi tiết các bước tiếp cận đến việc xây dựng một lược đồ dữ liệu đồ thị dựa trên lược đồ hình sao và bông tuyết. Bước đầu tiên cần phải thiết kế một mô hình dữ liệu đa chiều bao gồm các bảng sự kiện và các bảng chiều, tiếp theo sử dụng các quy luật chuyển đổi được đề xuất, thực hiện chuyển đổi các bảng sự kiện và bảng chiều sang các nút và các khóa ngoại của các bảng sẽ được chuyển thành cạnh, các chiều phân cấp của lược đồ bông tuyết cũng sẽ được chuyển thành nút và có mối liên kết với chiều cha của nó.

Sau đây hình 2.7, 2.8, 2.9, 2.10sẽ minh họa của việc chuyển đổi từ lược đồ hình sao và lược đồ bông tuyết sang mô hình dữ liệu đồ thị của tác giả.

**Table

Description automatically generated**

Hình . Lược đồ quan hệ hình sao ban đầu [4]

**Diagram

Description automatically generated**

Hình . Mô hình dữ liệu đồ thị sau khi được chuyển đồ từ mô hình dữ liệu hình sao [4]

**Table

Description automatically generated**

Hình . Lược đồ bông tuyết ban đầu [4]

**Diagram

Description automatically generated**

Hình . Lược đồ đồ thị sau khi chuyển đổi từ lược đồ bông tuyết [4]

Sau khi chuyển đổi từ mô hình dữ liệu đa chiều sang mô hình dữ liệu đồ thị, tác giả đưa ra các thí nghiệm hiệu suất để xem lược đồ nào sẽ có hiệu suất đọc/ghi tốt nhất. Tác giả thực hiện so sánh 4 loại lược đồ gồm:

1. Lược đồ dữ liệu quan hệ hình sao
2. Lược đồ dữ liệu quan hệ bông tuyết
3. Lược đồ dữ liệu đồ thị hình sao
4. Lược đồ dữ liệu đồ thị bông tuyết

Tác giả đã đề ra các câu truy vấn khác nhau để thực hiện thí nghiệm hiệu suất đọc/ghi của các lược đồ. Ở phần ghi dữ liệu vào kho dữ liệu, việc ghi dữ liệu của 2 loại lược đồ quan hệ có hiệu suất như nhau và chiếm thời gian thấp hơn 14 lần so với mô hình dữ liệu đồ thị, đặc biệt lược đồ dữ liệu đồ thị bông tuyết chiếm thời gian cao nhất khi dung lượng cần thiết để ghi dữ liệu ngày càng tăng lên. Lý do cho việc ghi dữ liệu của 2 lược đồ đồ thị cao hơn sao với lược đồ quan hệ là do cần thêm thời gian để tạo các mối liên kết giữa các nút.

Chart, line chart

Description automatically generated

Hình . Hiệu suất của các lược đồ khi thực hiện ghi dữ liệu [4]

Về hiệu suất của đọc dữ liệu (hình 2.11), các câu truy vấn không phân cấp (trong câu truy vấn các bảng chiều không có chứa các phân cấp của chúng) cho thấy rằng hiệu suất của 2 lược đồ quan hệ nhanh gấp 10 lần so với kho dữ liệu đồ thị do câu truy vấn đơn giản và không cần thực hiện phép kết bảng nhiều. Hình 2.12 là hai câu truy vấn không phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4] với bảng sự kiện là *Store\_Sales* và hai bảng chiều là *date\_dim* và *item*. Kết quả hiệu suất của hai câu truy vấn được thể hiện ở hình 2.13.

A picture containing text

Description automatically generated

Hình . Hai câu truy vấn không phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4]

Chart, line chart

Description automatically generated

Hình .. Hiệu suất của các lược đồ khi thực hiện truy vấn các dữ liệu không phân cấp [4]

Về hiệu suất của các câu truy vấn có phân cấp (trong câu truy vấn có các bảng chiều và các phân cấp của chúng), lược đồ dữ liệu đồ thị đã thể hiện ra ưu thế rõ rệt hơn so với lược đồ quan hệ (hơn 20 lần) do khi truy vấn lược đồ quan hệ cần thực hiện phép toán kết bảng giữa các bảng sự kiện và bảng chiều, với hiệu suất ngày càng tệ đi khi khối lượng dữ liệu cần truy vấn ngày càng lớn và chậm nhất trong 4 lược đồ là lược đồ quan hệ bông tuyết do số lượng kết bảng nhiều nhất. Hình 2.14 là hai câu truy vấn có phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4] với bảng sự kiện là *Store\_Sales* và các bảng chiều là *promotion, date\_dim, customer* cùng với phân cấp của customer là *customer\_demographics*. Kết quả hiệu suất của hai câu truy vấn được thể hiện ở hình 2.15.

Table

Description automatically generated with medium confidence

Hình . Hai câu truy vấn có phân cấp SQL và Cypher được sử dụng trong nghiên cứu [4]

Chart, line chart

Description automatically generated

Hình . Hiệu suất của các lược đồ khi thực hiện truy vấn các dữ liệu có phân cấp [4]

Từ các kết quả so sánh cho thấy khả năng ghi dữ liệu của mô hình dữ liệu quan hệ tốt hơn so với mô hình dữ liệu đồ thị, nhưng khi truy vấn dữ liệu, mô hình dữ liệu đồ thị lại áp đảo hơn so với mô hình dữ liệu quan hệ trong trường hợp truy vấn phân cấp, do mô hình dữ liệu đồ thị không cần các phép kết bảng giữa các bảng sự kiện với các bảng chiều và khi truy vấn càng nhiều phân cấp, ưu thế của mô hình dữ liệu đồ thị càng được thể hiện rõ ràng hơn. Nhưng khi truy vấn không phân cấp, lược đồ đồ thị lại có thời gian truy vấn chậm hơn so với lược đồ quan hệ, do khi câu truy vấn không quá phức tạp, lược đồ quan hệ không cần phải thực hiện kết bảng nhiều mà chỉ truy vấn các cột các hàng trong một bảng, trong khi GDW cần phải duyệt đồ thị để tìm kiếm các thực thể trong câu truy vấn để trả về kết quả. Cuối cùng, kết luận của công trình nghiên cứu đưa ra, tùy vào các trường hợp khác nhau, có thể sử dụng các lược đồ dữ liệu khác nhau để cài đặt hệ thống kho dữ liệu. Trường hợp của kho dữ liệu đồ thị phù hợp cho các loại dữ liệu có tính kết nối cao, phân cấp và phức tạp như dữ liệu mạng xã hội và xây dựng hệ thống tư vấn thời gian thực do khả năng truy vấn dữ liệu liên kết khá tốt và khi dữ liệu quá lớn, trong khi đó kho dữ liệu truyền thống với lược đồ hình sao và bông tuyết lại không thể đáp ứng tốt được.

### Kết luận và so sánh kiến trúc của kho dữ liệu đồ thị và kho dữ liệu quan hệ

Dựa trên các đặc điểm và cơ chế hoạt động của kho dữ liệu đồ thị và kho dữ liệu quan hệ truyền thống, chúng em xây dựng bảng so sánh đặc điểm cho hai loại kho dữ liệu theo Bảng 2.2 dưới đây.

Bảng . Bảng so sánh kiến trúc giữa kho dữ liệu đồ thị và kho dữ liệu quan hệ

|  |  |  |
| --- | --- | --- |
|  | **Kho dữ liệu đồ thị** | **Kho dữ liệu quan hệ** |
| Định dạng lưu trữ | Nút và cạnh. | Dòng và cột. |
| Mối quan hệ | Thể hiện bằng các cạnh giữa các nút. | Liên kết giữa các bảng, được thành lập bởi các khóa ngoại giữa các bảng với nhau. |
| Mô hình dữ liệu | Mô hình dữ liệu đồ thị. | Mô hình dữ liệu quan hệ đa chiều. |
| Cách thức thực thi câu truy vấn | Thực hiện phép toán duyệt đồ thị. Tất cả mọi thứ về hệ thống đều được tối ưu hóa để có thể duyệt hết dữ liệu một cách nhanh chóng. | Phép kết các bảng sự kiện và bảng chiều (và sẽ phức tạp hơn nếu là lược đồ bông tuyết). |
| Cơ chế thực hiện câu truy vấn của hệ thống | Mối quan hệ giữa các nút được lưu trữ sẵn trên ổ cứng, được đặt tên và định hướng sẵn để giúp cho việc duyệt đồ thị dễ dàng hơn. | Lúc kết bảng sẽ cần phải duyệt hết tất cả các bảng có trong câu truy vấn và tạo kết nối thời điểm chạy. |
| Cách lưu trữ dữ liệu dưới hệ thống | Chỉ ra kết nối ngay lúc dữ liệu được thêm vào để tránh vấn đề về hiệu suất tìm kiếm chỉ mục cho phép toán kết bảng (join index lookup performance) khi duyệt đồ thị. Điều này khiến đồ thị trở nên tối ưu khi siêu mô hình (meta-model) có nhiều mối quan hệ chồng chéo. | Lưu trữ liên kết giữa các bảng ở cấp độ logic và phải dựa trên đại số quan hệ để thay đổi dữ liệu được lưu trữ ở hệ quản trị CSDL có định dạng logic liên quan. |
| Độ linh hoạt | Rất linh hoạt, có thể thay đổi lược đồ tùy ý mà không cần thay đổi lại cả hệ thống kho dữ liệu. | Chỉ cần một thay đổi nhỏ về cột hoặc bảng có thể dẫn đến thay đổi toàn bộ hệ thống kho dữ liệu. |
| Đồng nhất về dữ liệu | Đồ sộ và không đồng nhất. | Đồng nhất về dữ liệu. |
| Loại OLAP (phân tích xử lý trực tuyến) | Graph-based OLAP (GOLAP). | Relational-OLAP (ROLAP), Multidimensional-OLAP (MOLAP) và Hybrid-OLAP (HOLAP). |
| Ngôn ngữtruy vấn | Không xác định, tùy vào DBMS khác nhau (như Cypher của Neo4j, NGQL của Nebula Graph, ...). | SQL. |
| Thời gian xuất hiện | Được đề xuất vào những năm 2011 [20], hiện tại đã xuất hiện nhiều nghiên cứu về kho dữ liệu đồ thị hơn. | Được đề ra vào 1990 bởi Bill Inmon [2], đến hiện nay đã được củng cố và vững chắc, được giới công nghiệp sử dụng rộng rãi. |

Qua nội dung các công trình nghiên cứu đề cập, cũng như bảng so sánh được lập ở trên, chúng ta có thể thấy được các điểm mạnh của kho dữ liệu đồ thị so với kho dữ liệu quan hệ truyền thống. Trong trường hợp lưu trữ dữ liệu quá lớn và khi cần truy vấn các dữ liệu có cấu trúc liên kết cao và phức tạp, chúng ta có thể áp dụng kho dữ liệu đồ thị vào các hệ thống bởi khả năng truy vấn và khai thác dữ liệu mạnh mẽ của cấu trúc dữ liệu đồ thị mang lại, không cần các phép toán kết bảng giữa bảng sự kiện và bảng chiều như kho dữ liệu truyền thống. Các nghiên cứu về kho dữ liệu đồ thị đã xuất hiện từ trước nhưng chưa được áp dụng rộng rãi vì vấn đề dữ liệu lớn chưa bùng nổ. Hiện tại khi hiện tượng dữ liệu lớn trở nên phổ biến thì các bài nghiên cứu về kho dữ liệu đồ thị nhiều hơn. Tuy nhiên vẫn chưa có một quy trình rõ ràng, vậy nên trong đề tài nghiên cứu này, chúng em sẽ đề xuất ra một quy trình để xây dựng hoàn thiện một hệ thống gồm GDW được thiết kế nhằm tối ưu việc lưu trữ dữ liệu kết hợp với các kho dữ liệu đồ thị theo nhóm chức năng (Graph Data Mart, viết tắt là GDM) để phục vụ trả lời cho các nhu cầu thống kê, phân tích xu hướng của nghề nghiệp, kỹ năng nghề nghiệp cũng như tư vấn các khóa học theo mục tiêu nghề nghiệp trong lĩnh vực CNTT. Với lượng dữ liệu lớn về kỹ năng, nghề nghiệp và khóa học trong bài toán được liên kết chặt chẽ, các câu truy vấn phục vụ cho thống kê và tư vấn thuộc dạng phân cấp từ đó thể hiện rõ được ưu thế về hiệu suất của kho dữ liệu đồ thị áp dụng cho bài toán. Cuối cùng chúng em thực hiện so sánh hiệu suất của hệ thống kho dữ liệu đồ thị với kho dữ liệu quan hệ truyền thống để kiểm chứng sự hiệu quả của hệ thống.

## Kết luận chương

*Trong chương này, chúng em đã, đưa ra hiện trạng về kho dữ liệu trong hiện tượng “dữ liệu lớn” hiện nay. Đồng thời, chúng em cũng sơ lược về CSDL đồ thị, giới thiệu và tổng quan các nghiên cứu về kho dữ liệu đồ thị, cuối cùng đưa ra bảng so sánh giữa kho dữ liệu truyền thống và kho dữ liệu đồ thị. Từ đó nêu ra kho dữ liệu đồ thị phù hợp để sử dụng làm kho dữ liệu trong phân tích, thống kê xu hướng nghề nghiệp, khóa học và lộ trình học tập cho mục nghề nghiệp.*

*Ở chương sau, chúng em sẽ phát biểu phạm vi bài toán hệ thống phân tích thống kê tư vấn nghề nghiệp và lộ trình học tập, mô tả các yêu cầu chức năng của bài toán. Cuối cùng, đưa ra phương pháp đề xuất về mô hình dữ liệu đồ thị và trình bày về cơ sở lý thuyết, lý luận cho phương pháp nghiên cứu được sử dụng trong khóa luận.*

**Chương 3**

# Kiến trúc dữ liệu đồ thị đề xuất

*Nội dung của chương này trình bày về phạm vị bài toán hệ thống phân tích thống kê tư vấn nghề nghiệp và lộ trình học tập, mô tả các yêu cầu chức năng, cơ sở xác định các lớp thực thể, thiết kế và mô tả kiến trúc đồ thị đề xuất. Cuối cùng phân tích cơ sở khoa học của giải pháp về mô hình đồ thị đưa ra cho bài toán.*

## Phạm vi bài toán

Lĩnh vực CNTT là một khái niệm rất rộng lớn, nó bao hàm nhiều chuyên ngành nhỏ bên trong. Bên cạnh đó các kỹ năng và kiến thức trong lĩnh vực này được cập nhật thường xuyên, đòi hỏi nhân lực phải có chuyên môn khá cao. Chính vì điều đó đối với một người theo học trong lĩnh vực CNTT, sẽ khó tránh khỏi việc phân vân trong quá trình lựa chọn hướng đi nghề nghiệp và các kỹ năng cần thiết cho bản thân để phù hợp với nghề nghiệp trong bối cảnh hiện tại. Theo khảo sát, trong năm 2019 Việt Nam thiếu đến 90.000 nhân lực, trong năm 2020 con số này tăng lên 400.000 và ước tính vào năm 2021 là 500.000 nhân lực [21]. Tình trạng thiếu hụt này xuất phát từ sự chênh lệch giữa trình độ của nhân lực và yêu cầu của doanh nghiệp, số lượng nhân lực có trình độ chuyên môn cao không đủ đáp ứng cho thị trường, trong khi đội ngũ nhân sự mới thì lại thiếu những kỹ năng cần thiết do sự thiếu định hướng trong quá trình học tập [21]. Chính vì thế, việc học tập và bổ sung thêm kiến thức và kỹ năng nghề nghiệp ngày càng trở nên quan trọng. Hiện nay, điều này đã dễ dàng hơn với sự hỗ trợ của các hệ thống E-Learning. Tuy nhiên việc lựa chọn và sắp xếp khóa học phù hợp cũng là một thách thức cho người học bởi vì có quá nhiều sự lựa chọn với một số lượng lớn khóa học được cung cấp từ nhiều nền tảng học trực tuyến như Udemy, Coursera, Linkedin, …. Để giải quyết các vấn đề về lựa chọn nghề nghiệp và lộ trình học tập, sự hỗ trợ đến từ một hệ thống tư vấn nghề nghiệp và đề xuất khóa học theo lộ trình học tập là một trong những phương án có khả năng đem lại hiệu quả tốt. Việc xây dựng hệ thống này giúp người học có thể định hướng và lựa chọn nghề nghiệp phù hợp với bản thân, bên cạnh đó nắm được các kỹ năng cần thiết phục vụ cho nghề nghiệp mong muốn thông qua các khóa học trực tuyến được tổ chức theo lộ trình hợp lý.

Để có thể xây dựng được một hệ thống tư vấn nghề nghiệp, lộ trình học tập hiệu quả thì một nguồn dữ liệu được tổ chức đủ tốt phục vụ cho việc khai thác và xử lý dữ liệu một cách liên tục và nhanh chóng là một yếu tố cần thiết. Chính vì điều đó, đề tài tiến hành nghiên cứu và xây dựng một hệ thống kho dữ liệu được sử dụng như một nguồn tri thức phục vụ phát triển hệ thống tư vấn nghề nghiệp và lộ trình học tập.

## Mô tả yêu cầu chức năng

Trong đề tài, chúng em nghiên cứu, phân tích và xây dựng kho dữ liệu đồ thị phục vụ cho các yêu cầu về tư vấn nghề nghiệp và lộ trình học tập cho người học trong phạm vi lĩnh vực CNTT. Trong đó dữ liệu được khai thác phục vụ cho các yêu cầu cụ thể của bài toán. Các yêu cầu được phân chia theo chức năng bao gồm chức năng phân tích thống kê, chức năng tìm kiếm và chức năng tư vấn. Trong các mục 3.2.1, 3.2.2 và 3.2.3 sẽ liệt kê, mô tả và phân tích các yêu cầu trong 3 nhóm chức năng trên.

### Nhóm chức năng phân tích thống kê

Với nhóm chức năng phân tích thống kê bao gồm 3 yêu cầu:

* Yêu cầu 1: Trong ngành CNTT, nghề nghiệp nào đang được ưa chuộng.
* Yêu cầu 2: Trong mỗi nghề nghiệp, kỹ năng nào cần thiết.
* Yêu cầu 3: Với mỗi kỹ năng, kỹ năng đó đang phổ biến trong nghề nghiệp nào.

Mô tả và phân tích các yêu cầu trong nhóm chức năng phân tích thống kê sẽ được trình bày trong các bảng 3.1, 3.2 và 3.3 dưới đây.

Bảng . Yêu cầu 1 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Trong ngành CNTT, nghề nghiệp nào đang được ưa chuộng. |
| **Mã yêu cầu** | YC1 |
| **Mô tả** | Nghề nghiệp đang được ưa chuộng, nghĩa là nghề nghiệp có số bài đăng tuyển dụng tăng lên nhiều nhất trên các nền tảng tuyển dụng.  Yêu cầu này giúp cho người dùng nắm được tình hình về nghề nghiệp trong lĩnh vực CNTT từ đó xem xét và lựa chọn nghề nghiệp phù hợp với bản thân và bối cảnh hiện tại. |
| **Kỹ thuật phân tích số liệu** | Tính số lượng bài đăng tuyển dụng của từng nghề nghiệp.  Thực hiện sắp xếp danh sách nghề nghiệp theo thứ tự giảm dần của số lượng bài đăng tuyển dụng nghề nghiệp đó. |

Bảng . Yêu cầu 2 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Trong mỗi nghề nghiệp, kỹ năng nào cần thiết. |
| **Mã yêu cầu** | YC2 |
| **Mô tả** | Với mỗi nghề nghiệp sẽ yêu cầu các kỹ năng khác nhau. Kỹ năng cần thiết là những kỹ năng được nhiều bài đăng tuyển dụng của nghề nghiệp đó yêu cầu/đề cập đến.  Thống kế này giúp người dùng nắm được kỹ năng nào cần thiết/quan trọng cho ngành nghề họ đang quan tâm, từ đó có thể có kế hoạch phát triển kỹ năng đúng đắn cho nghề nghiệp họ mong muốn. |
| **Kỹ thuật phân tích số liệu** | Với mỗi nghề nghiệp, tính số lượng bài đăng có yêu cầu theo mỗi kỹ năng. |

Bảng . Yêu cầu 3 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Với mỗi kỹ năng, kỹ năng đó đang phổ biến trong nghề nghiệp nào. |
| **Mã yêu cầu** | YC3 |
| **Mô tả** | Với mỗi kỹ năng sẽ phù hợp các một số nghề nghiệp khác nhau. Kỹ năng đang phổ biến trong ngành nghề nghĩa là nhiều bài đăng tuyển dụng về nghề nghiệp đó có yêu cầu kỹ năng đang xét.  Câu trả lời của YC3 sẽ giải đáp cho trường hợp khi người dùng biết về một kỹ năng nào đó, họ có thể sẽ đưa ra câu hỏi liệu kỹ năng này sẽ được áp dụng cho nghề nghiệp nào, từ đó lựa chọn nghề nghiệp theo kiến thức của bản thân. |
| **Kỹ thuật phân tích số liệu** | Với mỗi kỹ năng, tính số lượng bài đăng tuyển dụng về mỗi nghề nghiệp có yêu cầu kỹ năng đó. |

### Nhóm chức năng tìm kiếm

Với nhóm chức năng tìm kiếm bao gồm 1 yêu cầu:

* Yêu cầu 4: Tìm kiếm các khóa học được cung cấp bởi các nền tảng cung cấp khóa học trực tuyến mở đại trà (Massive Open Online Courses, viết tắt là MOOCs) dựa theo kỹ năng khóa học đào tạo.

Mô tả và phân tích các yêu cầu trong nhóm chức năng tìm kiếm sẽ được trình bày trong các bảng 3.4 dưới đây.

Bảng . Yêu cầu 4 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Tìm kiếm các khóa học được cung cấp bởi các nền tảng MOOCs  dựa theo kỹ năng khóa học đào tạo. |
| **Mã yêu cầu** | YC4 |
| **Mô tả** | Người dùng tìm kiếm khóa học về một kỹ năng, hệ thống sẽ đưa ra các khóa học đào tạo kỹ năng người dùng tìm kiếm.  Bên cạnh đó, người dùng có thể điều chỉnh bộ lọc bao gồm trình độ, ngôn ngữ, nền tảng MOOC cung cấp khóa học để tìm kiếm khóa học phù hợp. |
| **Kỹ thuật phân tích dữ liệu** | Thực hiện tìm kiếm các khóa học đào tạo kỹ năng người dùng tìm kiếm. Sắp xếp, hiển thị kết quả các khóa học đã tìm được theo thứ tự giảm dần của số người đăng ký khóa học và đánh giá. Ngoài ra, khi người dùng điều chỉnh bộ lọc để tìm kiếm khóa học phù hợp để tìm kiếm khóa học phù hợp, hệ thống truy vấn các dữ liệu liên quan đến khóa học bao gồm trình độ, ngôn ngữ, nền tảng MOOC cung cấp khóa học và trả về kết quả phù hợp. |

### Nhóm chức năng tư vấn

Cuối cùng, với nhóm chức năng tư vấn bao gồm 2 yêu cầu:

* Yêu cầu 5: Tư vấn các nghề nghiệp phù hợp cho người học dựa trên các kỹ năng đã có của họ.
* Yêu cầu 6: Đưa ra lộ trình học tập dựa trên các khóa học trên các nền tảng MOOCs cho các nhóm người học khác nhau.

Mô tả và phân tích các yêu cầu trong nhóm chức năng tư vấn sẽ được trình bày trong các bảng 3.5 và 3.6 dưới đây.

Bảng . Yêu cầu 5 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Tư vấn các nghề nghiệp phù hợp cho người học dựa trên các kỹ năng đã có của họ. |
| **Mã yêu cầu** | YC5 |
| **Mô tả** | Khi người dùng đăng ký tài khoản, sẽ cung cấp các kỹ năng đã có. Từ các kỹ năng đã có của người dùng và kỹ năng yêu cầu của nghề nghiệp, hệ thống đề xuất công việc phù hợp cho người dùng. |
| **Kỹ thuật phân tích số liệu** | Hệ thống trả về nghề nghiệp có nhiều kỹ năng mà người dùng đã có nhất. |

Bảng . Yêu cầu 6 của bài toán

|  |  |
| --- | --- |
| **Yêu cầu** | Đưa ra lộ trình học tập dựa trên các khóa học trên các nền tảng MOOCs cho các nhóm người học khác nhau. |
| **Mã yêu cầu** | YC6 |
| **Mô tả** | Đưa ra lộ trình học tập sẽ gồm các khóa học giảng dạy để phục vụ cho một nghề nghiệp, được lựa chọn phù hợp với kỹ năng ban đầu (nếu có) của người dùng.  Đối tượng người dùng bao gồm:   * Dành cho người bắt đầu từ đầu với một nghề nghiệp đã chọn. * Người đã biết một vài kiến thức, muốn học tiếp nghề nghiệp đã chọn. * Người muốn chuyển từ nghề nghiệp này sang nghề nghiệp khác. |
| **Kỹ thuật phân tích số liệu** | Đối với người bắt đầu từ đầu, mặc định người bắt đầu từ đầu thì không có bất kì kỹ năng nào. Đối với người có một vài kỹ năng tìm lộ trình phù hợp thì kết hợp với những kỹ năng người dùng sẵn có, và cấp độ của kỹ năng đó.  Các bước để đưa ra lộ trình học tập:   * Bước 1. Liệt kê danh sách các kỹ năng cần thiết trong nghề nghiệp. * Bước 2. Với mỗi kỹ năng, chọn khóa học liên quan có đánh giá và lượt đăng ký cao nhất. * Bước 3. Đưa ra tập các khóa học liên quan đến các kỹ năng ở bước 1. Tập khóa học được sắp xếp dựa vào kỹ năng đầu vào và đầu ra của các khóa học (nếu có). * Bước 4. Với những kỹ năng người học đã có, loại bỏ các khóa học liên quan trong tập các khóa học. |

## Cơ sở thiết kế kiến trúc đồ thị đề xuất

Để có thể phục vụ được các yêu cầu ở mục 3.2, kiến trúc đồ thị mới được xây dựng, trong đó chứa các lớp thực thể chính bao gồm:

* Nghề nghiệp (Career): Nghề nghiệp được đăng tuyển dụng trong những bài đăng trên các nền tảng tuyển dụng.
* Khóa học (Course): Các khóa học được cung cấp bởi các nền tảng học trực tuyến.
* Năng lực nghề nghiệp (Competency): Các kỹ năng mà các bài đăng tuyển dụng yêu cầu, cũng là các kỹ năng mà các khóa học trên các nền tảng học trực tuyến đào tạo.
* Người dùng (User): Khi người dùng đăng ký tài khoản trên hệ thống, hệ thống sẽ ghi nhận lại các thông tin người dùng cung cấp như tên, email đăng nhập, các kỹ năng đã có và nghề nghiệp mong muốn (nếu có).

Từ các lớp thực thể này, chúng em thiết kế kiến trúc đồ thị được kế thừa từ hai công trình nghiên cứu [6] và [9]. Mô hình kiến trúc đồ thị kế thừa sự phân tích các lớp thực thể khóa học, năng lực, nghề nghiệp với đặc trưng của nó theo các mô hình tiêu chuẩn nghề nghiệp hiện nay trong nghiên cứu [6]. Cụ thể, lớp khóa học bao gồm thực thể khóa học và các thông tin về khóa học như nền tảng (website), ngôn ngữ (language), người hướng dẫn (instructor) và trình độ (level). Trong lớp năng lực nghề nghiệp được tổng hợp và phân loại thành 5 nhóm theo các đặc điểm, bao gồm: Ngôn ngữ lập trình (programing language), công cụ hoặc phần mềm hỗ trợ lập trình (tool), bộ khung hỗ trợ lập trình (framework), các công nghệ được sử dụng làm nền tảng cho việc xây dựng và phát triển ứng dụng (platform) và kiến thức lý thuyết (knowledge). Bên cạnh đó, các yêu cầu trong chức năng tư vấn của đề tài hướng đến phục vụ cho việc cá nhân hóa người dùng, vậy nên mô hình thực hiện thừa kế lớp người dùng từ nghiên cứu [9].

**Diagram

Description automatically generated**

Hình . Kiến trúc đồ thị của nghiên cứu [6]

Diagram

Description automatically generated

Hình . Kiến trúc đồ thị của nghiên cứu [9]

Sau khi kế thừa sự phân tích về các lớp thực thể từ hai công trình nghiên cứu [6] [9] và tiến hành điều chỉnh để phù hợp với phạm vi yêu cầu bài toán. Trong đó, tiến hành lược bỏ các thực thể không liên quan đến các yêu cầu của bài toán như Industry, Taxonomy, Softskill, Certification, Organization, Program trong mô hình từ nghiên cứu [4]. Bên cạnh đó, để thực hiện thống kê xu hướng việc làm, từ mô hình gốc ban đầu tiến hành xây dựng thêm thực thể bài đăng tuyển dụng (Jobposting) trong lớp nghề nghiệp, thể hiện thông tin các bài đăng tuyển dụng. Từ đó, chúng em tiến hành đề xuất mô hình thiết kế kiến trúc đồ thị trong hình 3.3. Trong đó lớp người dùng sẽ có mối liên hệ với lớp nghề nghiệp và lớp năng lực nghề nghiệp để thể hiện mục tiêu nghề nghiệp và năng lực nghề nghiệp của người dùng đó từ đó thực hiện cá nhân hóa hệ thống tư vấn. Bên cạnh đó, để xây dựng được lộ trình học theo kĩ năng phục vụ cho mỗi nghề nghiệp, cần phải có nắm được kỹ năng yêu cầu của nghề nghiệp và kỹ năng được đào tạo trong các khóa học vậy nên cần mối quan hệ giữa lớp nghề nghiệp, lớp khóa học với lớp năng lực nghề nghiệp.

Diagram, schematic, bubble chart

Description automatically generated

Hình . Kiến trúc đồ thị đề xuất cho bài toán

Các mô tả chi tiết về thực thể và mối quan hệ trong kiến trúc đồ thị được liệt kê trong bảng 3.7, bảng 3.8 và bảng 3.9.

Bảng . Các thực thể trong kiến trúc đồ thị đề xuất

|  |  |  |
| --- | --- | --- |
| **Lớp** | **Thực thể** | **Mô tả thực thể** |
| User | User | Mô tả thông tin người dùng cung cấp khi đăng kí vào hệ thống. |
| Career | Career | Mô tả nghề nghiệp. |
| JobPosting | Các bài đăng tuyển dụng trên nền tảng tuyển dụng như Linkedin, TopCV, Glints, … |
| Course | Course | Khóa học trên các nền tảng học trực tuyến. |
| Level | Mức độ khó của khóa học. |
| Instructor | Người hướng dẫn khóa học. |
| Language | Ngôn ngữ mà khóa học hỗ trợ. |
| Website | Các trang nền tảng cung cấp khóa học trực tuyến như Coursera, Udemy, Linkedin Learning, … |
| Competency | Programming Language | Ngôn ngữ lập trình như C++, Python, Java, … |
| Tool | Công cụ hoặc phần mềm hỗ trợ lập trình như Visual Studio, GitHub, IntelliJ IDEA, … |
| Framework | Bộ khung hỗ trợ lập trình như .NET, Angularjs, Flutter, … |
| Platform | Nhóm các công nghệ được sử dụng làm nền tảng cho việc xây dựng và phát triển ứng dụng. |
| Knowledge | Kiến thức lý thuyết như Data Modeling, Object Oriented Programming – OOP, Index, … |

Bảng . Các thuộc tính của thực thể trong kiến trúc đồ thị đề xuất

|  |  |  |
| --- | --- | --- |
| **Thực thể** | **Thuộc tính** | |
| **Tên thuộc tính** | **Mô tả** |
| User | id: int | Id người dùng |
| name: string | Tên người dùng |
| email: string | Email đăng nhập của người dùng. |
| Career | name: string | Tên của nghề nghiệp. |
| Jobposting | id: int | Id của bài đăng tuyển dụng. |
| timestamp: string | Thời điểm đổ dữ liệu vào DW. |
| Course | id: int | Id định danh của khóa học. |
| name: string | Tên của khóa học. |
| enrolled: integer | Số lượng người đăng ký khóa học tại thời điểm thu thập dữ liệu. |
| url: string | Đường dẫn đến khóa học. |
| time: string | Thời gian học. |
| fee: string | Học phí tại thời thu thập dữ liệu. |
| rating: float | Điểm đánh giá tại thời điểm thu thập dữ liệu. |
| timestamp: string | Thời gian nút được tạo. |
| Level | name: string | Tên của mức độ khó của khóa học. |
| Instructor | name: string | Tên của người hướng dẫn khóa học. |
| Language | name: string | Tên ngôn ngữ mà khóa học hỗ trợ. |
| Website | name: string | Tên nền tảng cung cấp khóa học. |
| ProgrammingLanguage | name: string | Tên ngôn ngữ lập trình. |
| Tool | name: string | Tên công cụ hoặc phần mềm hỗ trợ lập trình. |
| Framework | name: string | Tên bộ khung hỗ trợ lập trình. |
| Platform | name: string | Tên nền tảng hỗ trợ cho việc xây dựng và phát triển ứng dụng. |
| Knowledge | name: string | Tên kiến thức lý thuyết. |

Bảng . Các mối quan hệ trong kiến trúc đồ thị đề xuất

|  |  |  |
| --- | --- | --- |
| **Mối liên kết** | **Tên cạnh liên kết** | **Mô tả** |
| User – Career | FAVOR\_IN | Thể hiện nghề nghiệp mà người dùng mong muốn. |
| User – Competency | HAVE\_PL | Thể hiện các ngôn ngữ lập trình mà người dùng đã biết. |
| HAVE\_TOOL | Thể hiện các công cụ mà người dùng đã biết. |
| HAVE\_FRAM | Thể hiện các khung lập trình mà người dùng đã biết. |
| HAVE\_PLAT | Thể hiện các nền tảng mà người dùng đã biết. |
| HAVE\_KNOW | Thể hiện các kiến thức lý thuyết mà người dùng đã biết. |
| JobPosting – Career | ABOUT | Thể hiện bài đăng tuyển dụng này thuộc ngành nghề nào. |
| Job Posting – Competency | NEED\_PL | Thể hiện công việc này cần biết ngôn ngữ lập trình nào. |
| NEED\_TOOL | Thể hiện công việc này cần biết sử dụng công cụ nào. |
|  |
| NEED\_FRAM | Thể hiện công việc này cần biết các sử dụng khung lập trình nào. |  |
|  |
| NEED\_PLAT | Thể hiện công việc này cần các nền tảng nào. |  |
| NEED\_KNOW | Thể hiện công việc này cần kiến thức nào. |  |
| NEED\_CERT | Thể hiện công việc này cần chứng chỉ CNTT nào. |  |
| Course – Competency | TEACH\_PL | Thể hiện khóa học này cung cấp ngôn ngữ lập trình nào. |  |
| TEACH\_TOOL | Thể hiện khóa học này cung cấp công cụ nào. |  |
| TEACH\_FRAM | Thể hiện khóa học này cung cấp khung lập trình nào. |  |
| TEACH \_PLAT | Thể hiện khóa học này cung cấp nền tảng nào. |  |
| TEACH \_KNOW | Thể hiện khóa học này cung cấp kiến thức nền tảng nào. |  |
| REQUIRE\_PL | Thể hiện ngôn ngữ lập trình đầu vào mà khóa học yêu cầu. |  |
| REQUIRE\_TOOL | Thể hiện công cụ đầu vào mà khóa học yêu cầu. |  |
| REQUIRE\_FRAM | Thể hiện khung lập trình đầu vào mà khóa học yêu cầu. |  |
| REQUIRE\_PLAT | Thể hiện nền tảng đầu vào mà khóa học yêu cầu. |  |
| REQUIRE\_KNOW | Thể hiện kiến thức nền tảng đầu vào mà khóa học yêu cầu. |  |
| Course – Level | HAS\_LEVEL | Thể hiện mức độ khó của khóa học. |  |
| Course – Instructor | INSTRUCT\_BY | Thể hiện khóa học được hướng dẫn bởi ai. |  |
| Course – Language | HAVE\_LANG | Thể hiện khóa học được giảng dạy bằng ngôn ngữ hoặc có phụ đề bằng ngôn ngữ gì. |  |
| Course – Website | PROVIDE\_BY | Thể hiện khóa học được cung cấp bởi trang web nền tảng giảng dạy trực tuyến nào. |  |

Ở hình 3.4 là một ví dụ cụ thể được áp dụng vào kiến trúc đồ thị đã đề xuất.

Diagram

Description automatically generated

Hình . Ví dụ cụ thể áp dụng kiến trúc đồ thị đề xuất

Sau khi người dùng “An” hoàn tất quá trình đăng ký tài khoản, dữ liệu người dùng cung cấp bao gồm thông tin cá nhân, năng lực nghề nghiệp đã có. Thông tin cá nhân như tên và email sẽ được lưu trữ trong các thuộc tính của thực thể người dùng. Các năng lực đã có bao gồm ngôn ngữ lập trình “python” và kiến thức “ETL” được biểu diễn bằng liên kết giữa người dùng và các nút kỹ năng tương ứng. Người dùng có thể cung cấp nghề nghiệp họ quan tâm lúc đăng ký vào hệ thống hoặc điền thông tin sau này. Khi nghề nghiệp người dùng mong muốn được cung cấp, cụ thể trong ví dụ là “data engineer”, mối liên kết giữa thực thể người dùng “An” và nút nghề nghiệp “data engineer” sẽ được hình thành. Với nghề nghiệp “data engineer” bao gồm 2 bài đăng tuyển dụng. Bài đăng “JP1” yêu cầu 3 kỹ năng nghề nghiệp bao gồm ngôn ngữ lập trình Python, công cụ “apache spark” và kiến thức về “ETL”. Bài đăng “JP2” yêu cầu 2 kỹ năng nghề nghiệp là ngôn ngữ lập trình “python”, và kiến thức về “ETL”. Trong dữ liệu về khóa học bao gồm 2 khóa học. Khóa học “C1” dạy về công cụ” apache spark” với điều kiện đầu vào biết về ngôn ngữ “python”, khóa học ở mức độ sơ cấp (beginner) được giảng dạy bằng ngôn ngữ tiếng Việt do người hướng dẫn Bình giảng dạy trên nền tảng trực tuyến “Udemy”. Khóa học “C2” dạy về kiến thức “ETL” không yêu cầu kiến thức đầu vào, khóa học được giảng dạy bởi “John” bằng ngôn ngữ tiếng Anh (English) và cũng được cung cấp trên nền tảng “Udemy”.

Từ những yêu cầu của bài toán trong đề tài được mô tả trong mục 3.2, có thể thấy hầu hết các yêu cầu đều thuộc dạng truy vấn phân cấp vậy nên việc truy vấn thông qua các phép kết bảng ở mô hình dữ liệu quan hệ sẽ khiến hiệu suất truy vấn không hiệu quả. Một hệ thống tư vấn thời gian thực đòi hỏi cần có một mô hình dữ liệu đủ tốt với lượng dữ liệu được cập nhật liên tục và hiệu suất truy vấn cao vậy nên việc sử dụng mô hình quan hệ sẽ không được tối ưu cho bài toán. Bên cạnh đó, với mối liên kết chặt chẽ giữa các lớp thực thể và số lượng dữ liệu khổng lồ của các khóa học trên nhiều nền tảng học trực tuyến và các bài đăng ở các trang tuyển dụng khiến hiệu suất của việc truy vấn trên mô hình dữ liệu đồ thị càng trở lên áp đảo hơn. Ngoài ra, đối với hệ thống tư vấn, có nhiều phương pháp tiếp cận và xử lý sử dụng đến cấu trúc dữ liệu dạng đồ thị, như xử lý tính toán dưới dạng đồ thị và áp dụng các thuật toán khai thác đồ thị [22] [23] [24] vì vậy việc áp dụng kiến trúc đồ thị vào hệ thống xây dựng kho dữ liệu có thể giúp hỗ trợ cho việc cài đặt mô hình tư vấn sau này. Với những lợi ích này, có thể thấy rằng, việc áp dụng cấu trúc dữ liệu đồ thị cho việc xây dựng hệ thống kho dữ liệu đồ thị phục vụ cho bài toán tư vấn nghề nghiệp và lộ trình học tập trong đề tài là hợp lý.

## Kết luận chương

*Ở chương này chúng em đã trình bày về phạm vi của bài toán, các chức năng của bài toán, thiết kế kiến trúc đồ thị để giải quyết các yêu cầu của bài toán và phân tích cơ sở khoa học về giải pháp đưa ra cho đề tài. Ở chương sau, chúng em phân tích các thành phần của kiến trúc và lựa chọn các công nghệ cho mỗi thành phần.*

**Chương 4**

# Thiết kế kiến trúc tổng quát của hệ thống

*Trong chương này, nội dung trình bày về thiết kế kiến trúc tổng quan của hệ thống. Tiếp theo, mô tả giải pháp cho các yêu cầu của bài toán và thiết kế các kho dữ liệu theo nhóm chức năng tương ứng trong kiến trúc tổng quan.*

## Thiết kế kiến trúc tổng quát của hệ thống

Với các yêu cầu đã được nêu ra trong mục 3.2, để xây dựng kho dữ liệu với mô hình đồ thị được thiết kế trong hình 3.3, hệ thống cần tập dữ liệu về nghề nghiệp, khóa học, năng lực nghề nghiệp và người dùng. Dữ liệu cần được thu thập, xử lý và đổ vào GDW từ đó xây dựng các GDM để phục vụ cho các yêu cầu của bài toán. Vì vậy, mô hình kiến trúc cần bao gồm các lớp như sau:

* Lớp dữ liệu nguồn (Data Source Layer): Thành phần cung cấp dữ liệu thô (trang khóa học, nền tảng tuyển dụng, trang của hệ thống tư vấn).
* Lớp thu thập dữ liệu (Data Acquisition Layer): Thành phần thu thập dữ liệu từ nơi cung cấp dữ liệu thô.
* Lớp lưu trữ dữ liệu thô (Data Lake Layer): Thành phần lưu trữ thô sau khi được thu thập.
* Lớp xử lý dữ liệu (Data Processing Layer): Thành phần xử lý dữ liệu bao gồm mô hình rút trích dữ liệu, làm sạch dữ liệu và đổ dữ liệu vào GDW.
* Lớp kho dữ liệu (Data Warehouse Layer): Thành phần lưu trữ dữ liệu đã được xử lý, dữ liệu được lưu trữ dưới dạng đồ thị.
* Lớp kho dữ liệu theo nhóm chức năng (Data Mart Layer): Thành phần phân chia lưu trữ dữ liệu theo các nhóm chức năng phục vụ cho từng yêu cầu của bài toán.
* Lớp ứng dụng dữ liệu (Data Serving Layer): Thành phần tính năng của bài toán, sử dụng dữ liệu có được để phục vụ cho các yêu cầu này.

Toàn bộ quy trình của hệ thống được chúng em thực hiện xây dựng theo kiến trúc được thiết kế trong hình 4.1.

Diagram

Description automatically generated

Hình . Kiến trúc tổng quát của hệ thống

Trong kiến trúc của hệ thống, đầu vào của hệ thống GDW sẽ là các nguồn dữ liệu từ khóa học của các trang khóa học trực tuyến, các bài đăng tuyển dụng trên các nền tảng tuyển dụng và dữ liệu người dùng được hệ thống ghi nhận khi người dùng cung cấp thông tin trên ứng dụng của bài toán. Các dữ liệu thô này sau khi được lấy về sẽ đưa vào kho dữ liệu thô để lưu trữ, sau đó sẽ sử dụng công cụ xử lý dữ liệu lớn để thực hiện xử lý và làm sạch trước khi đưa vào GDW. Trong quá trình xử lý và làm sạch dữ liệu, các dữ liệu thô đa phần đều được lưu trữ dưới dạng tệp tin, trong đó, các tệp tin về khóa học và bài đăng tuyển dụng đều có một cột chứa mô tả về khóa học, mô tả về những kỹ năng nghề nghiệp CNTT sẽ được hướng dẫn trong khóa học đó, hoặc chứa mô tả về các kỹ năng CNTT yêu cầu trong công việc đó. Để có thể trích xuất ra được thông tin về các kỹ năng nghề nghiệp trong đoạn văn bản mô tả đó, hệ thống sẽ cần một mô hình rút trích thực thể (Named-Entity Recognition, viết tắt là NER) để rút trích các cụm từ liên quan tới kỹ năng nghề nghiệp. Mô hình NER này sẽ được kế thừa từ công trình nghiên cứu [6], hình 4.2 minh họa một đoạn văn bản mô tả của một khóa học, các cụm từ được in đậm và gạch dưới là những từ sẽ được mô hình rút trích thực thể nhận diện và sau đó gán nhãn tùy theo kỹ năng nghề nghiệp.

Text

Description automatically generated

Hình . Ví dụ minh họa cho việc rút trích thực thể trong văn bản từ mô hình rút trích thực thể

Sau khi dữ liệu được xử lý và làm sạch, hệ thống tiến hành đổ chúng vào GDW để lưu trữ. Cuối cùng, dữ liệu trong GDW sẽ được truy vấn, xử lý và đưa vào các GDM để phục vụ những yêu cầu trong các nhóm chức năng ở mục 3.1.2 của bài toán. Tương ứng với ba nhóm chức năng bao gồm chức năng phân tích thống kê, chức năng tìm kiếm và chức năng tư vấn, chúng em đề xuất xây dựng ba GDM bao gồm: Kho dữ liệu phân tích thống kê (Statistical Analysis Data Mart, viết tắt là  SDM) được sử dụng để phục vụ cho các yêu cầu trong nhóm chức năng phân tích thống kê, kho dữ liệu khóa học (Course Data Mart, viết tắt là CDM) phục vụ cho yêu cầu trong nhóm chức năng tìm kiếm, đồng thời kết hợp SDM, CDM với kho dữ liệu người dùng (User Data Mart, viết tắt là UDM) và kho dữ liệu liên kết các khóa học (Course-Course Data Mart, viết tắt CCDM) để cung cấp dữ liệu cho yêu cầu trong nhóm chức năng tư vấn bao gồm tư vấn nghề nghiệp và tư vấn lộ trình bao gồm các khóa học cho các đối tượng người dùng. Cơ sở thiết kế cấu trúc của các DM sẽ được trình bày chi tiết trong mục 4.2.

## Mô hình GDM trong kiến trúc của hệ thống

Để thiết kế được các mô hình của GDM trong kiến trúc của hệ thống, trong mục 4.2.1, 4.2.2 và 4.2.3 chúng em tiến hành mô tả giải pháp từ phân tích các yêu cầu trong ba nhóm chức năng ở mục 3.2 để hình thành cơ sở thiết kế GDM cho hệ thống.

### Giải pháp cho nhóm chức năng phân tích thống kê

Với nhóm chức năng phân tích thống kê trong mục 3.2.1, các giải pháp đã đưa vào mô hình thiết kế SDM được mô tả trong bảng 4.1 sau.

Bảng . Giải pháp thiết kế DM cho nhóm chức năng phân tích thống kê

|  |  |  |
| --- | --- | --- |
| **Mã yêu cầu** | **Phân tích yêu cầu** | **Giải pháp** |
|  |
| YC1 | Tính số lượng bài đăng tuyển dụng của từng nghề nghiệp. | Trong thực thể nghề nghiệp (Career) khởi tạo thuộc tính *numpost* là số bài đăng tuyển dụng về nghề nghiệp tương ứng tại các thời gian thu thập dữ liệu. Mỗi lần truy vấn dữ liệu, câu truy vấn được viết sẽ chọn ra nút Career và xét thuộc tính thời gian để thực hiện tính toán. |  |
| YC2 | Với mỗi nghề nghiệp, tính số lượng bài đăng có yêu cầu theo mỗi kỹ năng. | Với 5 thành phần kỹ năng, tạo các biến *numPl*, *numTool*, *numFw*, *numPf*, *numKl* là thuộc tính của các mối liên kết nối từ thực thể nghề nghiệp đến các nút kỹ năng, mỗi thuộc tính là số lượng bài đăng tuyển dụng về nghề nghiệp đó có yêu cầu kỹ năng tương ứng. |  |
|  |
|  |
|  |
|  |
| YC3 | Với mỗi kỹ năng, tính số lượng bài đăng tuyển dụng về mỗi nghề nghiệp có yêu cầu kỹ năng đó. | Câu hỏi tương tự chỉ ngược vế với YC2. Sử dụng lại các biến *numPl*, *numTool*, *numFw*, *numPf*, *numKl* nhưng xét theo chiều ngược lại là từ kỹ năng sang nghề nghiệp. |  |

Các yêu cầu trong nhóm chức năng phân tích thống kê đều liên quan đến thực thể nghề nghiệp và năng lực vậy nên trong mô hình kiến trúc của SDM bao gồm hai nhóm thực thể là nghề nghiệp và các kỹ năng trong nhóm năng lực. Với các thuộc tính *numPl*, *numTool*, *numFw*, *numPf*, *numKl* trong các mối liên kết giữa thực thể nghề nghiệp và 5 kỹ năng tương ứng trong nhóm năng lực thể hiện số lượng bài đăng tuyển dụng về một nghề nghiệp và có yêu cầu các kỹ năng tương ứng. Từ các giải pháp trên, chúng em xây dựng SDM phục vụ nhóm chức năng phân tích thống kê với thiết kế trong hình 4.3.

Bubble chart

Description automatically generated

Hình . Mô hình thiết kế SDM

Bảng . Mô tả thành phần thực thể trong SDM

|  |  |  |
| --- | --- | --- |
| **Thực thể** | **Thuộc tính** | **Mô tả** |
| Career | name: string  numpost: int  timestamp: string | Career: Thực thể nghề nghiệp.   * name: tên của nghề nghiệp. * numpost: số bài đăng tuyển dụng của mỗi nghề nghiệp tại thời gian thu thập. * timestamp: thời điểm nút được tạo. |
| Programing Language | name: string | Programing Language: Ngôn ngữ lập trình nghề nghiệp yêu cầu.   * name: tên ngôn ngữ lập trình. |
| Tool | name: string | Tool: Công cụ hoặc phần mềm hỗ trợ lập trình nghề nghiệp yêu cầu.   * name: tên công cụ. |
| Framework | name: string | Framework: Bộ khung hỗ trợ lập trình nghề nghiệp yêu cầu.   * name: tên bộ khung. |
| Platform | name: string | Platform: Nền tảng xây dựng và phát triển ứng dụng nghề nghiệp yêu cầu.   * name: tên nền tảng. |
| Knowledge | name: string | Knowledge: Kiến thức lý thuyết nghề nghiệp yêu cầu.   * name: tên kiến thức. |

### Giải pháp cho nhóm chức năng tìm kiếm

Với nhóm chức năng tìm kiếm, giải pháp đưa vào mô hình DM được mô tả trong bảng 4.3 sau.

Bảng . Giải pháp thiết kế DM cho nhóm chức năng tìm kiếm

|  |  |  |
| --- | --- | --- |
| **Mã yêu cầu** | **Phân tích yêu cầu** | **Giải pháp** |
| YC4 | Thực hiện tìm kiếm các khóa học đào tạo kỹ năng người dùng tìm kiếm. Sắp xếp, hiển thị kết quả các khóa học đã tìm được theo thứ tự giảm dần của người đăng ký khóa học và đánh giá. Ngoài ra, khi người dùng điều chỉnh bộ lọc để tìm kiếm khóa học phù hợp để tìm kiếm khóa học phù hợp, hệ thống truy vấn các dữ liệu liên quan đến khóa học bao gồm trình độ, ngôn ngữ, nền tảng MOOC cung cấp khóa học và trả về kết quả phù hợp. | Để thực hiện tìm kiếm và hiển thị thông tin khóa học, hệ thống cần được cung cấp các dữ liệu về khóa học như tên khóa học, lượt đăng ký, đánh giá, học phí, url, người hướng dẫn và các kỹ năng mà khóa học đào đạo. Bên cạnh đó cần các thông tin liên quan để áp dụng bộ lọc như độ khó, ngôn ngữ, trang nền tảng cung cấp khóa học. |

Yêu cầu trong nhóm chức năng tìm kiếm liên quan đến các dữ liệu về khóa học vậy nên trong CDM bao gồm các thực thể khóa học, nền tảng cung cấp khóa học, mức độ khó, người hướng dẫn, ngôn ngữ giảng dạy và các kỹ năng khóa học giảng dạy. Từ đó, chúng em xây dựng CDM phục vụ nhóm chức năng tìm kiếm với thiết kế trong hình 4.4.

Diagram

Description automatically generated

Hình . Mô hình thiết kế CDM

Bảng . Mô tả thành phần thực thể trong CDM

|  |  |  |
| --- | --- | --- |
| **Thực thể** | **Thuộc tính** | **Mô tả** |
| Course | id: int  name: string  enrolled: int  url: string  time: string  fee: string  rating: float  timestamp: string | Course: một thực thể khóa học   * id: id định danh của khóa học. * name: tên của khóa học. * enrolled: số lượng người đăng ký khóa học tại thời điểm hiện tại. * url: đường dẫn đến khóa học. * time: thời gian học. * fee: học phí tại thời điểm hiện tại. * rating: đánh giá tại thời điểm hiện tại. * timestamp: thời gian nút được tạo khi ở GDW. |
| Website | name: string | Website: website cung cấp khóa học   * name: tên website. |
| Level | name: string | Level: mức độ khó của khóa học   * name: tên của mức độ. |
| Instructor | name: string | Instructor: người hướng dẫn khóa học   * name: tên của người hướng dẫn. |
| Language | name: string | Language: Ngôn ngữ khóa học hỗ trợ   * name: tên ngôn ngữ. |
| Programing Language | name: string | Programing Language: Ngôn ngữ lập trình khóa học giảng dạy.   * name: tên ngôn ngữ lập trình.. |
| Tool | name: string | Tool: Công cụ hoặc phần mềm hỗ trợ lập trình khóa học giảng dạy.   * name: tên công cụ. |
| Framework | name: string | Framework: Bộ khung hỗ trợ lập trình khóa học giảng dạy.   * name: tên bộ khung. |
| Platform | name: string | Platform: Nền tảng xây dựng và phát triển ứng dụng khóa học giảng dạy.   * name: tên nền tảng. |
| Knowledge | name: string | Knowledge: Kiến thức lý thuyết khóa học giảng dạy.   * name: tên kiến thức. |

### Giải pháp cho nhóm chức năng tư vấn

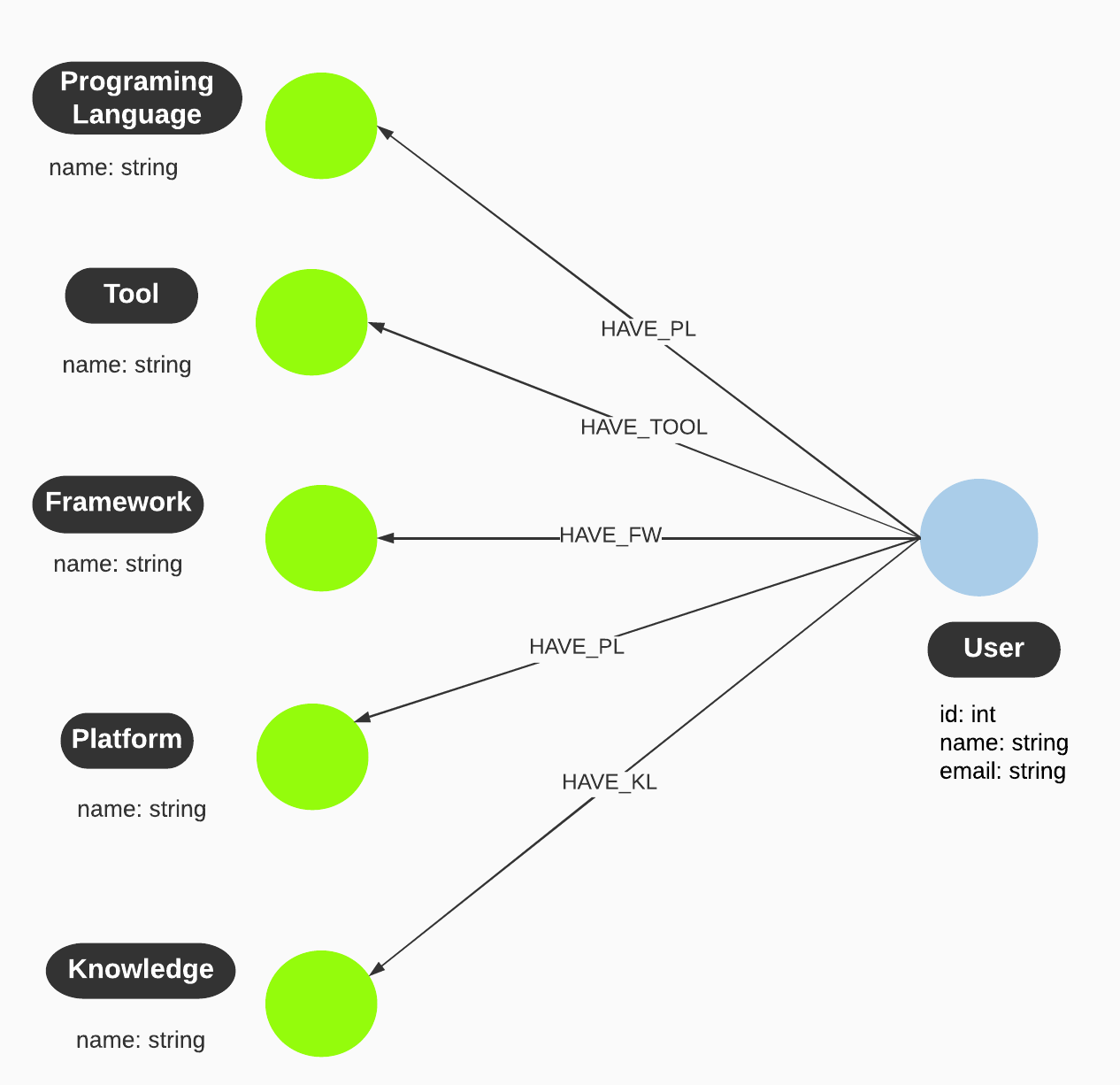
Với nhóm chức năng tư vấn, các giải pháp đã đưa vào mô hình thiết kế được mô tả trong bảng 4.5 sau.

Bảng . Giải pháp thiết kế DM cho nhóm chức năng tư vấn

|  |  |  |
| --- | --- | --- |
| **Mã yêu cầu** | **Phân tích yêu cầu** | **Giải pháp** |
| YC5 | Hệ thống trả về nghề nghiệp có nhiều kỹ năng mà người dùng đã có nhất. | Để thực hiện yêu cầu, cần xác định các dữ liệu của người dùng như thông tin tài khoản và những kỹ năng người dùng đã có. Kết hợp với SDM để có dữ liệu về các kỹ năng mà nghề nghiệp yêu cầu.  → Tạo một DM gồm dữ liệu của người dùng. |
| YC6 | Đối với người bắt đầu từ đầu, mặc định người bắt đầu từ đầu thì không có bất kì kỹ năng nào. Đối với người có một vài kỹ năng tìm lộ trình phù hợp thì kết hợp với những kỹ năng người dùng sẵn có, và cấp độ của kỹ năng đó.  Các bước để đưa ra lộ trình học tập:   * Bước 1. Liệt kê danh sách các kỹ năng cần thiết trong nghề nghiệp. * Bước 2. Với mỗi kỹ năng, chọn khóa học liên quan có đánh giá và lượt đăng ký cao nhất. * Bước 3. Đưa ra tập các khóa học liên quan đến các kỹ năng ở bước 1. Tập khóa học được sắp xếp dựa vào kỹ năng đầu vào và đầu ra của các khóa học (nếu có). * Bước 4. Với những kỹ năng người học đã có, loại bỏ các khóa học liên quan trong tập các khóa học. | Để liệt kê danh sách các kỹ năng cần thiết trong nghề nghiệp, mô hình sử dụng dữ liệu từ SDM. Tiếp theo, sử dụng dữ liệu từ CDM để chọn các khóa học phù hợp có liên quan đến các kỹ năng yêu cầu. Sau khi có tập các khóa học, để sắp xếp theo trình tự, chúng em xây dựng CCDM để thể hiện sự liên quan giữa các khóa học, mục tiêu khi đưa vào một tập các khóa học cần học thì có thể đưa ra được thứ tự của nó nếu có, trong đó các khóa học được liên kết với nhau bằng cách  xét kỹ năng đầu vào của khóa học, sau đó liên kết các khóa học có đào tạo kỹ năng đầu vào mà khóa học đang xét yêu cầu với khóa học đó.  Ở bước 4, kết hợp kết quả nhận được ở các bước trên với dữ liệu trong UDM để loại bỏ các khóa học mà đào tạo kỹ năng người dùng đã có. |

Từ các giải pháp trên, chúng em tiến hành xây dựng mô hình của hai DM bao gồm UDM với kiến trúc trong hình 4.5 và CCDM với kiến trúc trong hình 4.6.

Trong UDM, cung cấp dữ liệu của người dùng bao gồm các thông tin cơ bản như tên, email và các kỹ năng mà người dùng đã có.



Hình . Mô hình thiết kế UDM

Bảng . Mô tả thành phần thực thể trong UDM

|  |  |  |
| --- | --- | --- |
| **Thực thể** | **Thuộc tính** | **Mô tả** |
| User | id: int  name: string  email: string | User: Thực thể người dùng   * id: id người dùng. * name: tên người dùng. * email: email đăng nhập. |
| Programing Language | name: string | Programing Language: Ngôn ngữ lập trình người dùng đã có.   * name: tên ngôn ngữ lập trình. |
| Tool | name: string | Tool: Công cụ hoặc phần mềm hỗ trợ lập trình người dùng đã có.   * name: tên công cụ. |
| Framework | name: string | Framework: Bộ khung hỗ trợ lập trình người dùng đã có.   * name: tên bộ khung. |
| Platform | name: string | Platform: Nền tảng xây dựng và phát triển ứng dụng người dùng đã có.   * name: tên nền tảng. |
| Knowledge | name: string | Knowledge: Kiến thức lý thuyết người dùng đã có.   * name: tên kiến thức. |

Trong CCDM, cung cấp trình tự các khóa học để phục vụ cho YC6, trong đó các khóa học được liên kết với nhau bằng cách xét kỹ năng đầu vào của khóa học, sau đó liên kết các khóa học có đào tạo kỹ năng đầu vào mà khóa học đang xét yêu cầu với khóa học đó.

Diagram

Description automatically generated

Hình . Mô hình thiết kế CCDM

Bảng . Mô tả thành phần thực thể trong CCDM

|  |  |  |
| --- | --- | --- |
| **Thực thể** | **Thuộc tính** | **Mô tả** |
| Course | id: int | Course: Thực thể khóa học   * id: id khóa học. |

## Kết luận chương

*Trong chương này, chúng em đã hoàn thành thiết kế kiến trúc logic của hệ thống, thiết kế các DM theo các nhóm chức năng của bài toán.* *Ở chương sau, chúng em tiến hành phân tích và lựa chọn các công nghệ để triển khai kiến trúc logic của hệ thống đã thiết kế.*

**Chương 5**

# Phân tích lựa chọn công nghệ triển khai hệ thống

*Nội dung chương này bao gồm phân tích các nhóm công nghệ trong kiến trúc triển khai. Từ đó lựa chọn các công nghệ thích hợp với yêu cầu của bài toán để áp dụng triển khai vào hệ thống.*

## Nhóm công nghệ trong kiến trúc triển khai

Với các thành phần trong kiến trúc logic ở mục 4.2, hệ thống cần những nhóm công nghệ tương ứng để tiến hành triển khai thực thi mô hình. Để thực hiện thu thập và lưu trữ dữ liệu từ lớp dữ liệu nguồn sang lớp lưu trữ dữ liệu thô, cần sử dụng công cụ cào dữ liệu (crawling service) để thu thập dữ liệu, dữ liệu sau khi được thu thập sẽ  được lưu dưới dạng các tập tin. Trong đề tài nghiên cứu này, chúng em thực hiện kế thừa tập dữ liệu thô đã được thu thập từ công trình nghiên cứu [6] và [25]. Sau đó, tiến hành tập hợp và lưu trữ tất cả dữ liệu trong hệ thống tập tin cục bộ đóng vai trò là một kho dữ liệu thô.

Với mục đích DW được sử dụng bởi hệ thống tư vấn thời gian thực đòi hỏi dữ liệu cần được cập nhật và xử lý một cách tự động và nhanh chóng, bên cạnh đó số lượng dữ liệu về khóa học, nghề nghiệp lại khá lớn. Vậy nên để hệ thống đáp ứng được cần áp dụng các công nghệ xử lý dữ liệu lớn trong lớp xử lý dữ liệu để tối ưu hóa khả năng xử lý dữ liệu. Hơn nữa, trong quá trình xử lý dữ liệu cần áp dụng mô hình rút trích thực thể, vậy nên công nghệ xử lý cần hỗ trợ cung cấp các tính năng để thực thi mô hình. Quá trình phân tích và lựa chọn công cụ xử lý dữ liệu lớn cho lớp xử lý dữ liệu sẽ được mô tả trong mục 5.2.2.

Cuối cùng, trong lớp kho dữ liệu và lớp kho dữ liệu theo nhóm chức năng, thực hiện lưu trữ dữ liệu đã được làm sạch vào GDW và GDM với mô hình dữ liệu phù hợp bằng các công nghệ lưu trữ CSDL đồ thị để phục vụ dữ liệu cho các yêu cầu của bài toán. Mục 5.2.1 sẽ trình bày phân tích lựa chọn công nghệ lưu trữ CSDL đồ thị cho hệ thống.

## Phân tích lựa chọn công nghệ

### Công nghệ lưu trữ CSDL đồ thị

Qua các ưu điểm của mô hình dữ liệu đồ thị được nêu ra ở mục 3.3 và sự phát triển của nó bởi nhu cầu sử dụng tăng cao, nhiều hệ quản trị CSDL đồ thị (Graph Database Management System, viết tắt là GDBMS), đã ra đời nhằm phục vụ cho việc lưu trữ, phân tích và xử lý với dữ liệu đồ thị. Qua hình 5.1 cho ta thấy được xu hướng của các GDBMS trong 10 năm gần đây.

Chart, line chart

Description automatically generated

Hình . Biểu đồ xu hướng của các hệ quản trị CSDL đồ thị [26]

Đã có nhiều GDBMS ra đời cho tới hiện nay, tùy vào các DBMS khác nhau mà có tích hợp các công cụ hay thư viện khác nhau nhằm phục vụ công việc phân tích dữ liệu đồ thị thuận lợi hơn so với chỉ lưu trữ và truy vấn thông thường. Người dùng sẽ dựa vào các yếu tố khác nhau của các nền tảng để lựa chọn GDBMS phù hợp. Từ thống kê trên hình 5.1 và thời gian phát triển của các GDBMS, chúng em thực hiện phân tích một số GDBMS để thực hiện chọn phương án thích hợp cho bài toán tư vấn của đề tài. Bảng đối sánh phân tích các GDBMS trong bảng 5.1 bao gồm các GDBMS ra đời sớm, có nhiều người sử dụng như Neo4j, Microsoft Azure Cosmos DB, ArangoDB và GDBMS ra đời sau và tốc độ tăng trưởng cao như TigerGraph, NebulaGraph.

Bảng . Bảng đối sánh giữa các CSDL đồ thị [9]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **GDBMS** | **Mô tả** | **Mô hình dữ liệu chính** | **Thuật toán trên đồ thị** | **Mô hình đồ thị gốc** | **Mã nguồn mở** | **Ngôn ngữ truy vấn** |
| **TigerGraph** | Nền tảng tính toán đồ thị hoàn chỉnh, phân tán, song song và hỗ trợ phân tích dữ liệu thời gian thực  là mô tả hệ quản trị CSDL nào. | CSDL đồ thị | TigerGraph Graph Data Science Library | Có | Không | GSQL |
| **Neo4j** | CSDL đồ thị có thể mở rộng, tuân thủ ACID được thiết kế với kiến trúc cụm phân tán cao, có sẵn trong dịch vụ tự lưu trữ của đám mây. | CSDL đồ thị | Neo4j Graph Data Science Library | Có | Có | Cypher |
| **ArangoDB** | Hệ thống CSDL đồ thị gốc, là CSDL đa mô hình. | Đa mô hình, bao gồm: lưu trữ dạng tài liệu JSON, khóa-giá trị, đồ thị. | ArangoDB NetworkX | Có | Có | SQL |
| **Microsoft Azure Cosmos DB** | Dịch vụ CSDL của Microsoft Azure hỗ trợ lưu trữ đa mô hình, các máy chủ được phân bố khắp nơi trên thế giới, có thể thoải mái mở rộng theo chiều ngang. | Đa mô hình, bao gồm: lưu trữ dạng tài liệu, đồ thị, khóa-giá trị, cột . | Không hỗ trợ thuật toán để xử lý đồ thị. | Không | Không | SQL |
| **Nebula Graph** | CSDL đồ thị phân tán, cung cấp khả năng mở rộng tuyến tính, có hiệu suất cao. | CSDL đồ thị | Spark GraphX | Có | Có | nGQL |

Dựa vào bảng 5.1, chúng em thấy rằng TigerGraph và Neo4j là hai GDBMS lý tưởng cho việc xây dựng hệ thống tư vấn theo thời gian thực. Cả hai GDBMS này đều là GDBMS lưu trữ gốc phổ biến, giúp tối ưu hóa tốc độ của các thuật toán trên đồ thị bởi vì dữ liệu được lưu trữ trực tiếp dưới dạng đồ thị. Ngoài ra, cả hai đều cung cấp thư viện khoa học dữ liệu đồ thị, giúp người dùng có thể sử dụng các thuật toán trên đồ thị một cách tiện lợi. Quan trọng hơn, tốc độ phản hồi của cả hai GDBMS đều rất nhanh, có tính khả dụng cao và mở rộng lớn, những yếu tố này đều phù hợp cho bài toán xây dựng hệ thống tư vấn thời gian thực. Tuy nhiên, TigerGraph lại tốn kém chi phí rất nhiều so với Neo4j, vậy nên, việc lựa chọn Neo4j làm CSDL đồ thị cho việc nghiên cứu của đề tài sẽ hiệu quả hơn về mặt chi phí. Ngoài ra, Neo4j còn là nền tảng có mặt từ rất sớm so với TigerGraph, trải qua nhiều phiên bản và cập nhật đã khiến Neo4j mang lại nhiều điểm vượt trội và ít bị lỗi hơn. Neo4j còn là nền tảng phân tích đồ thị được sử dụng nhiều nhất trong các CSDL đồ thị, cộng đồng người dùng rất lớn nên có thể dễ dàng tìm kiếm lỗi và sửa lỗi. Từ những lợi ích được nêu ra trên, nhóm chúng em sẽ lựa chọn Neo4j làm GDBMS để lưu trữ GDW và GDM cho đề tài.

### Công nghệ xử lý dữ liệu lớn

Hiện nay, giới công nghiệp không chỉ muốn đáp ứng được nhu cầu lưu trữ dữ liệu lớn, mà còn muốn truy vấn và xử lý dữ liệu với tốc độ cao. Với các yêu cầu về công nghệ xử lý dữ liệu được nêu ở mục 5.1, các công cụ xử lý dữ liệu truyền thống không còn thích hợp với nhu cầu của bài toán. Nhiều ứng dụng và công cụ xử lý dữ liệu lớn được công bố để đưa vào phục vụ giúp cho việc xử lý và phân tích các tập dữ liệu lớn trở nên đơn giản hơn, tăng tốc độ xử lý dữ liệu lớn từ đó có thể đưa ra kết quả sớm nhất cho người dùng khi truy vấn dữ liệu. Chúng cung cấp các cơ chế để phân tán dữ liệu, xử lý dữ liệu song song, và tận dụng tài nguyên của nhiều máy tính để tăng tốc độ xử lý dữ liệu. Một nghiên cứu được thực hiện bởi công ty dịch vụ đám mây Qubole vào năm 2020 [27] đã khảo sát sự sử dụng của các công cụ xử lý dữ liệu lớn phổ biến. Theo báo cáo này, Apache Spark là công cụ phổ biến nhất, được sử dụng bởi 71% số khảo sát. Tiếp theo đến là Apache Hadoop với tỷ lệ sử dụng là 41%, tiếp đến là Apache Hive với tỷ lệ 18%. Các công cụ khác như Apache Flink và MongoDB đều có tỷ lệ sử dụng thấp hơn, dưới 10%. Mỗi công nghệ có ưu điểm và hạn chế riêng tùy thuộc vào mục đích của bài toán sử dụng, vậy nên chúng em tiến hành lập bảng 5.2 so sánh để chọn công nghệ phù hợp với yêu cầu của bài toán.

Bảng . So sánh các công nghệ hỗ trợ xử lý dữ liệu lớn [28] [29] [30] [31] [32]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên công cụ** | **Hỗ trợ xử lý** | **Tốc độ xử lý** | **Ngôn ngữ lập trình** | **Hỗ trợ học máy** |
| **Apache Spark** | Batch, Streaming | Nhanh | Java, Python, Scala, R | Có |
| **Apache Hadoop** | Batch | Chậm | Java, Python, Scala | Không |
| **Apache Hive** | Batch | Chậm | SQL-like language | Không |
| **Apache Flink** | Batch, Streaming | Nhanh | Java, Scala | Có |
| **MongoDB** | Batch | Nhanh | MongoDB Query Language | Có |

Với yêu cầu về tư vấn thời gian thực đòi hỏi công nghệ hỗ trợ xử lý trực tuyến với tốc độ nhanh chóng, bên cạnh đó để áp dụng mô hình nhận diện thực thể công nghệ sử dụng phải hỗ trợ các tính năng về học máy. Vậy nên, dựa trên bảng 5.2 có thể thấy Apache Spark và Apache Flink là hai công nghệ hỗ trợ tốt nhất cho hệ thống của bài toán. Trong hai công nghệ này, Apache Spark được sử dụng rộng rãi hơn vậy nên cộng đồng phát triển và hỗ trợ lớn giúp cho việc cài đặt hệ thống trở nên dễ dàng hơn, Spark còn có tính linh hoạt trong việc lựa chọn cách triển khai (trên cụm hoặc trên nền tảng đám mây) và hỗ trợ nhiều ngôn ngữ lập trình. Và đặc biệt với việc lựa chọn Neo4j làm GDBMS cho hệ thống, Apache Spark lại càng là một công nghệ phù hợp cho quá trình xử lý với các tính năng tương tác với cơ sở dữ liệu đồ thị Neo4j. Cụ thể, Spark có thể đọc và ghi dữ liệu từ Neo4j thông qua Spark Neo4j connector. Spark Neo4j Connector cho phép Spark truy vấn dữ liệu trực tiếp từ Neo4j và xử lý dữ liệu đồ thị theo cách phù hợp với ứng dụng của mình. Với tính năng này, Spark cho phép tích hợp các ứng dụng phân tích dữ liệu của mình với dữ liệu đồ thị được lưu trữ trong Neo4j. Và với nhu cầu áp dụng mô hình rút trích thực thể trong kiến trúc, tồn tại một thư viện mã nguồn được mở xây dựng trên Apache Spark là Spark NLP. Thư viện được phát triển bởi John Snow Labs, để hỗ trợ triển khai xử lý ngôn ngữ tự nhiên (Natural Language Processing, viết tắt là NLP) với khả năng xử lý dữ liệu văn bản lớn với nhiều ngôn ngữ khác nhau. Ngoài ra, thư viện này còn hỗ trợ các tính năng tùy chỉnh, cho phép người dùng tạo các trình xử lý NLP tùy chỉnh và tích hợp chúng vào đề tài của mình [33]. Do đó, trong đề tài này, chúng em chọn Apache Spark làm công cụ để xử lý dữ liệu lớn cho xây dựng hệ thống GDW.

### Áp dụng công nghệ vào kiến trúc của hệ thống

Sau khi lựa chọn công nghệ ở mục 5.2.1 và 5.2.2, chúng em áp dụng công nghệ vào kiến trúc của hệ thống trong hình 5.2. Chi tiết mô tả công nghệ được áp dụng vào hệ thống sẽ được trình bày ở mục 6.1.

Diagram

Description automatically generated

Hình . Áp dụng công nghệ vào kiến trúc của hệ thống

## Kết luận chương

*Ở chương này, sau khi phân tích chúng em đã lựa chọn được các công nghệ phù hợp với kiến trúc và yêu cầu của hệ thống. Ở chương sau, chúng em tiến hành thiết kế triển khai, cài đặt thử nghiệm và phân tích các kết quả đạt được.*

**Chương 6**

# Thiết kế triển khai và cài đặt thử nghiệm

*Nội dung của chương mô tả kiến trúc triển khai cụ thể, mô tả chi tiết quá trình chạy dữ liệu trên hệ thống kho dữ liệu đồ thị, đưa ra kết quả thử nghiệm và so sánh kết quả với hệ thống kho dữ liệu truyền thống. Cuối cùng phân tích kết quả thử nghiệm và đưa ra nhận xét.*

## Triển khai hệ thống

Từ mô hình thiết kế kiến trúc của hệ thống được đưa ra ở mục 5.2 và kết quả nghiên cứu về kiến trúc hệ thống GDW, chúng em tiến hành áp dụng và đưa vào xây dựng cài đặt. Hình 6.1 là thiết kế chi tiết cho kiến trúc để thực hiện xây dựng thử nghiệm và Bảng 6.1 liệt kê chi tiết các công nghệ được sử dụng xây dựng hệ thống.

Graphical user interface, application

Description automatically generated

Hình . Sơ đồ kiến trúc chi tiết các công nghệ sử dụng xây dựng thử nghiệm hệ thống GDW trên nền tảng dữ liệu lớn

Bảng . Danh sách các công nghệ được sử dụng xây dựng hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Công nghệ** | **Vai trò trong hệ thống** | **Mô tả vai trò trong hệ thống** |
| 1 | Apache Spark 3.2.3 | Xử lý lô, Xử lý luồng | Các khung lập trình sử dụng:   * Spark SQL: dùng để xử lý dữ liệu được lấy từ hệ thống tập tin trong máy (local). * Spark Structured Streaming: xử lý các dữ liệu luồng từ tập tin User và đổ vào GDW cũng như chuyển dữ liệu từ GDW sang UDM. * Spark NLP 4.2.8: Tải mô hình NER, sau đó sử dụng mô hình vừa tải để nhận diện các thực thể kỹ năng nghề nghiệp từ các đoạn văn bản trong khung dữ liệu (dataframe), sau đó trả về một khung dữ liệu bao gồm các thực thể và loại thực thể đó thuộc về. * Neo4j Spark Connector: Dùng để kết nối với CSDL Neo4j. |
| 2 | Neo4j 5.4.0 (local) | GDW | Lưu trữ dữ liệu đồ thị sau khi được xử lý. |
| 3 | AuraDB (cloud) | GDM | Lưu trữ GDM cho từng bài toán nhỏ khác nhau của đề tài. |
| 4 | Docker | Container | Neo4j sẽ được cài đặt bằng docker. |
| 5 | PowerBI | Trực quan hóa dữ liệu | Tạo các dashboard trực quan hóa dữ liệu. |

### Hệ thống tập tin lưu trữ dữ liệu thô

Hệ thống sử dụng hệ thống tập tin ở máy chủ để lưu trữ dữ liệu thô tạm thời cho thử nghiệm lần này. Các tập tin sẽ lưu trữ trong các tệp được chia theo chủ đề riêng biệt để có thể dễ dàng quản lý. Toàn bộ dữ liệu thô sẽ được lưu trữ dưới định dạng .*csv*. Chi tiết về hệ thống tập tin được liệt kê dưới Bảng 6.2. Bảng 6.3 sẽ mô tả về cấu trúc dữ liệu trong dữ liệu thô về từng loại dữ liệu về khóa học, bài đăng tuyển dụng và người dùng trước khi chúng được đưa vào Spark để xử lý.

Bảng . Các tệp lưu trữ các tập tin trong hệ thống lưu trữ

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tệp** | **Chủ đề lưu trữ** |
| 1 | Course | Lưu trữ các dữ liệu thô về khóa học ở các trang tuyển dụng. |
| 2 | JobPost | Lưu trữ các dữ liệu thô về các bài đăng tuyển dụng. |
| 3 | User | Lưu trữ các dữ liệu thô về người dùng. |
| 4 | NERModel\_config | Lưu trữ mô hình nhận diện thực thể và các thông tin cấu hình của nó. |

Bảng . Mô tả về các cột được lưu trữ trong các dữ liệu thô khóa học, bài đăng tuyển dụng và người dùng

|  |  |  |
| --- | --- | --- |
| **STT** | **Tập tin thô ví dụ** | **Nội dung lưu trữ** |
| 1 | Coursera.csv | * name: Tên khóa học * link: Đường dẫn tới khóa học * rating: điểm đánh giá của khóa học (thang điểm 5) * enroll: Số người đã đăng ký khóa học * instructor: Người hướng dẫn khóa học * time: Thời gian để hoàn thành khóa học (giờ) * levelrequirement: Yêu cầu cấp độ đầu vào * skillrequirement: Kỹ năng đầu vào yêu cầu * skillWillLearn: Kỹ năng sẽ được giảng dạy trong khóa học * skillGain: Kỹ năng sẽ đạt được trong khóa học * subject: Lĩnh vực khóa học giảng dạy * organization: Tổ chức * fee: Phí đăng ký khóa học * program: Chương trình của khóa học * relationInsOrg: quan hệ của người hướng dẫn với tổ chức * subtitle: Các ngôn ngữ/phụ đề được sử dụng trong khóa học |
| 2 | Linkedin.csv | * name: Tiêu đề nghề nghiệp tuyển dụng * description: Mô tả của bài đăng tuyển dụng * salary: Mức lương chi trả |
| 3 | User.csv | * id: Mã định danh của người dùng * username: Tên người dùng * email: Email người dùng * career: Nghề nghiệp người dùng mong muốn * skills: Các kỹ năng nghề nghiệp thuộc lĩnh vực CNTT mà người dùng đã có |

### Sơ đồ xử lý dữ liệu và đổ dữ liệu vào kho dữ liệu đồ thị

Khung dữ liệu trong Apache Spark là một cấu trúc dữ liệu dạng bảng được phát triển để xử lý dữ liệu lớn, được tối ưu hóa cho hoạt động phân tán trên một cụm máy. Spark cung cấp API để người lập trình có thể truy vấn và thao tác dữ liệu với khung dữ liệu, giúp cho việc xử lý dữ liệu dễ dàng hơn [34].

Trong các đường ống dữ liệu được xây dựng, các dữ liệu khóa học, bài đăng tuyển dụng sau khi được thu thập về từ các trang mạng xã hội sẽ được xử lý theo lô. Các khung dữ liệu được xây dựng để đổ vào GDW sẽ tương ứng với các thực thể trong kiến trúc đồ thị được xây dựng ở Chương 3 (Hình 3.3). Đồng thời cũng xây dựng các khung dữ liệu quan hệ giữa hai thực thể để Spark Neo4j Connector có thể ánh xạ các quan hệ của chúng và xây dựng liên kết giữa hai thực thể đó. Hình 6.2 mô tả chi tiết quá trình xử lý dữ liệu của hai loại dữ liệu đó.

Text

Description automatically generated with low confidence

Hình . Sơ đồ cách xử lý dữ liệu khóa học và bài đăng tuyển dụng trong Spark

Mô hình NER trong đề tài sử dụng được kết thừa từ công trình nghiên cứu [6], là mô hình BERT với pretrain bert-base-uncased, được cung cấp bởi Huggingface thông qua thư viện Deep Learning transformer và được nhóm tác giả huấn luyện để nhận diện các cụm từ liên quan tới kỹ năng nghề nghiệp trong CNTT. Mô hình sẽ được áp dụng và tải vào thư viện Spark NLP, trong quá trình xử lý, đầu vào mô hình nhận diện sẽ đưa khung dữ liệu chứa trường dữ liệu cần được rút trích thực thể vào mô hình. Sau khi nhận diện thành công, Spark NLP sẽ trả về một khung dữ liệu chứa các trường, trong đó là mảng (array) cấu trúc annotatorType khác nhau. Các trường được sau khi rút trích thực thể được Spark NLP trả về được liệt kê chi tiết trong Bảng 6.4.

Bảng . Bảng chi tiết các trường được trả về từ khung dữ liệu của Spark NLP [33]

|  |  |  |
| --- | --- | --- |
| **STT** | **Trường kết quả** | **Ý nghĩa** |
| 1 | annotatorType | Loại chú thích đã tạo cho chú thích hiện tại. |
| 2 | begin | Điểm bắt đầu của các từ trong văn bản thô (Ví dụ từ hello world, điểm bắt đầu của hello là 0, world sẽ là 6). |
| 3 | end | Điểm kết thúc của các từ trong văn bản thô (Ví dụ từ hello world, điểm kết thúc của hello là 4, world sẽ là 10). |
| 4 | result | Kết quả chính của chú thích. |
| 5 | metadata | Nội dung của kết quả các từ được gán nhãn và các thông tin bổ sung. |
| 6 | embeddings | Xuất hiện trong phiên bản 2.0, chứa các vector ánh xạ (nếu cần). |

Kết quả sau khi nhận diện sẽ chứa các thông tin về từng từ trong đoạn văn bản đưa vào và đã được gán nhãn, tuy nhiên khung dữ liệu này vẫn chưa lấy được kết quả tường minh mong muốn, nên cần phải xử lý, gom nhóm các thông tin trong đó để đưa về kết quả mình mong muốn. Các nhãn thực thể trong mô hình có thể nhận diện bao gồm:

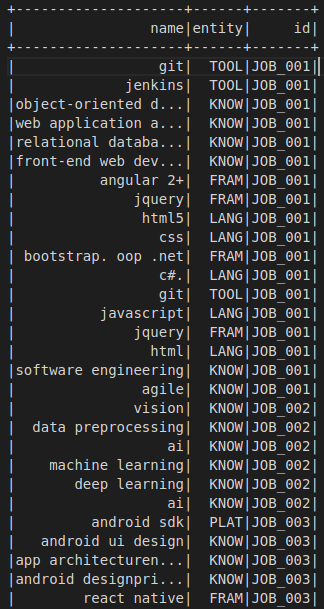
* LANG: Ngôn ngữ lập trình.
* TOOL: Công cụ lập trình.
* FRAM: Khung lập trình.
* PLAT: Nền tảng lập trình.
* KNOW: Kiến thức trong CNTT.

Hình 6.3 là ví dụ cho khung dữ liệu chứa trường dữ liệu đoạn văn bản cần đưa vào mô hình nhận diện thực thể (đoạn văn bản mô tả của các bài đăng tuyển dụng), và Hình 6.4 là kết quả trả khung dữ liệu đã qua xử lý và trả về khung dữ liệu có các trường gồm *name* là các thực thể nhận diện được trong các đoạn văn bản, *entity* là nhãn được gán cho thực thể đó và *id* là mã định danh để xác định kỹ năng yêu cầu đó thuộc về bài đăng tuyển dụng nào.

Graphical user interface, text

Description automatically generated

Hình . Khung dữ liệu đầu vào



Hình . Kết quả được trả về sau khi nhận diện thực thể

Tuy nhiên, mô hình cũng có nhiều điểm thiếu sót khi áp dụng rút trích thực thể. Như trên hình 6.4, ở dòng dữ liệu JOB\_001 của trường dữ liệu id, mô hình nhận diện bị lỗi khi nhận diễn cả 3 kỹ năng nghề nghiệp “bootstrap”, “oop”, “.net” thành 1 loại kỹ năng nghề nghiệp, chứ không phải là tách riêng thành 3 loại kỹ năng nghề nghiệp khác nhau. Hoặc trong 1 đoạn văn bản, cụm từ kỹ năng đó được đặt ở cuối câu và dính liền cới dấu chấm câu, mô hình sẽ rút trích kèm luôn cả cấu chấm câu vào và tính luôn là 1 loại kỹ năng nghề nghiệp. Cũng trường hợp như từ “sql” và “structured query language” thì “sql” sẽ được nhận diện như lả 1 kỹ năng nghề nghiệp, nhưng “structured query language” thì không được nhận diện như một kỹ năng. Tuy nhiên, trong trường hợp khác như “css” với “cascading style sheets” hay “html” với “hyper markup language” thì mô hình nhận diện được chúng và cũng cùng nhãn loại kỹ năng nghề nghiệp như nhau.

Về việc xử lý dữ liệu trùng lặp, Neo4j hỗ trợ xử lý dữ liệu đã tồn tại trong CSDL bằng câu lệnh *Merge*, câu lệnh *Merge* sẽ tìm kiếm trong CSDL đã tồn tại dữ liệu đó chưa, nếu đã có rồi thì Neo4j sẽ không tạo nút mới hoặc nếu có sự thay đổi về dữ liệu của nút đã tồn tại thì Neo4j sẽ tiến hành cập nhật lại nút đó. Khác hoàn toàn với câu lệnh *Create* khi chỉ đơn thuần tạo một nút mới mà không hề quan tâm dữ liệu đó đã tồn tại trong CSDL chưa. Spark Neo4j Connector hỗ trợ việc kết nối giữa Spark với CSDL Neo4j và hỗ trợ việc đọc/ghi dữ liệu nhanh chóng. Spark Neo4j Connector cung cấp các cấu hình lựa chọn để có thể thao tác việc đọc ghi với Neo4j tùy ý. Bảng 6.5 liệt kê các cấu hình trong đề tài sử dụng khi đọc/ghi dữ liệu với Neo4j với Spark Neo4j Connector.

Bảng . Các mô tả cấu hình trong Spark Neo4j Connector trên hệ thống thử nghiệm [35]

|  |  |  |
| --- | --- | --- |
| **STT** | **Cấu hình** | **Mô tả** |
| 1 | *mode* | Cách thức lưu trữ khi ghi dữ liệu vào Neo4j, gồm 2 loại:   * Overwrite: tương ứng với câu lệnh Merge. * Append: tương ứng với câu lệnh Create (Nếu sử dụng Spark dưới phiên bản 3.0 thì thay bằng ErrorIfExists). |
| 6 | *batch.size* | Số lượng dòng theo sẽ được chia nhỏ theo lô trong khung dữ liệu và xử lý, sau đó ghi vào Neo4j. |
| 7 | *node.keys* | Hoạt động giống như “khóa chính”, khi ghi vào Neo4j với hình thức “Overwrite”, cần xác định thuộc tính định danh trước khi thực hiện việc ghi, để Neo4j có thể tìm kiếm trong CSDL xem dữ liệu này đã tồn tại chưa. |
| 8 | *labels* | Nhãn của thực thể mà người dùng muốn đọc/ghi dữ liệu. |
| 9 | *relationship* | Tên mối quan hệ mà người dùng muốn ghi dữ liệu vào Neo4j. |
| 10 | *relationship.properties* | Các thuộc tính trong quan hệ cần ghi. |
| 11 | *relationship.save.strategy* | Chiến lược lưu trữ của mối quan hệ. |
| 12 | *relationship.source.labels* | Thực thể nguồn mà người dùng muốn tạo mối quan hệ. |
| 13 | *relationship.source.node.keys* | Điều kiện để ánh xạ với dữ liệu nguồn. |
| 14 | *relationship.source.save.mode* | Cách thức để ánh xạ với dữ liệu nguồn (Mặc định là “*Match*”). |
| 15 | *relationship.target.labels* | Thực thể đích mà người dùng muốn tạo mối quan hệ. |
| 16 | *relationship.target.node.keys* | Điều kiện để ánh xạ với dữ liệu đích. |
| 17 | *relationship.target.save.mode* | Cách thức ánh xạ với dữ liệu đích (Mặc định là “*Match*”). |
| 18 | *streaming.property.name* | Thuộc tính để đọc dữ liệu theo lô (dùng trong xử lý luồng). |
| 19 | *streaming.from* | Xác định thời điểm dữ liệu để Spark đọc dữ liệu từ thời điểm đó, gồm 2 lựa chọn:   * NOW: Bắt đầu đọc dữ liệu từ thời điểm luồng bắt đầu. * ALL: Đọc hết dữ liệu từ Neo4j trước khi đọc các dữ liệu mới từ luồng. |
| 20 | *streaming.query.offset* | Một câu lệnh chỉ đọc (READ\_ONLY)  Cypher sẽ trả về giá trị long. Ví dụ *(Match (p:MyLabel) return Max(p.timestamp))*. Câu lệnh vừa đưa ra được dùng để lấy các dữ liệu mới nhất trong Neo4j. Cấu hình này được sử dụng khi người dùng muốn đọc dữ liệu bằng câu truy vấn tùy ý trong xử lý luồng. |
| 21 | *checkpointLocation* | Đường dẫn đến tệp tin cho phép Spark Structured Streaming phục hồi khi bị lỗi. Spark cập nhật tệp này với thông tin tiến trình và khôi phục từ thời điểm đó trong trường hợp phục hồi lỗi hoặc thời điểm câu truy vấn bắt đầu (Sử dụng khi ghi luồng). |
| 18 | *query* | Câu truy vấn tùy ý của người dùng muốn sử dụng khi đọc/ghi. |

Với cấu hình *batch.size*, dữ liệu trong khung dữ liệu sẽ được đổ vào CSDL Neo4j theo lô. Khi dữ liệu được xử lý và đổ vào Neo4j, *batch.size* sẽ giúp chia nhỏ dòng dữ liệu và xử lý theo lô với kích thước lô theo ý muốn, sau đó đổ vào Neo4j thay vì xử lý toàn bộ trước rồi mới đổ vào. Điều này giúp tăng hiệu suất và giảm tải bộ nhớ sử dụng của máy.

Các dữ liệu không có nhãn thời gian sẽ sử dụng cách thức lưu trữ *mode* là *Overwrite*, tương đương với câu lệnh *Merge* trong Neo4j. Riêng thực thể Course và Jobposting được gán nhãn thời gian sẽ lưu trữ mode là *Append*, tương đương với câu lệnh *Create*, có nghĩa là sẽ tạo mới các nút khi đổ dữ liệu vào Neo4j. Điều này thể hiện sự thay đổi của dữ liệu theo nhãn thời gian để phù hợp với bài toán thống kê theo xu hướng được nêu ra ở chương 4, mục 4.2.1.

Đối với dữ liệu người dùng, các bước xử lý dữ liệu sẽ đơn giản hơn do không cần sử dụng mô hình rút trích thực thể do dữ liệu đầu vào đã có gán nhãn trước cho các kỹ năng nghề nghiệp. Hình 6.5 là chi tiết quá trình xử lý dữ liệu của người dùng.

Diagram

Description automatically generated with low confidence

Hình . Sơ đồ cách xử lý dữ liệu User trong Spark

### Sơ đồ xử lý chuyển dữ liệu từ GDW sang GDM

Ở chương 4, đề tài đã xây dựng 4 kiến trúc GDM tương ứng với cho các yêu cầu khác nhau của bài toán (xem lại mục 4.2). Các hình 6.6, 6.7, 6.8, 6.9 là các chi tiết quá trình xử lý chuyển dữ liệu từ GDW vào 4 GDM, tương ứng với các GDM là SDM, CDM, UDM và CCDM (hình 4.3, 4.4, 4.5, 4,6). Quá trình chuyển dữ liệu từ GDW sang GDM đều có điểm chung là truy vấn dữ liệu từ GDW, sau đó tùy theo từng yêu cầu khác nhau sẽ có sự tùy chỉnh các mối quan hệ trong đồ thị con.

#### Sơ đồ chuyển dữ liệu từ GDW sang SDM

Với SDM (Hình 6.6), khi truy vấn lấy dữ liệu từ GDW, sẽ cần phải truy vấn ra đồ thị con với các thực thể và liên kết gồm Career, Jobposting, các thực thể Competency và các mối quan hệ giữa chúng (bước 1). Sau đó, đếm số lượng bài đăng tuyển dụng liên quan đến Career đó,  gán thêm thuộc tính vào thực thể Career (bước 2) và sau đó đổ vào SDM (bước 3). Tiếp theo, đếm tần suất yêu cầu kỹ năng nghề nghiệp về nghề nghiệp đó dựa trên số lượng bài đăng liên quan đến nghề nghiệp đó (bước 4). Sau đó tạo mối quan hệ giữa Career và các thực thể Competency và gắn thuộc tính trên mối quan hệ là số tần suất vừa tìm được (bước 5), cuối cùng đổ hết tất cả các khung dữ liệu vừa được xây dựng vào SDM (bước 6) (hình 6.6).

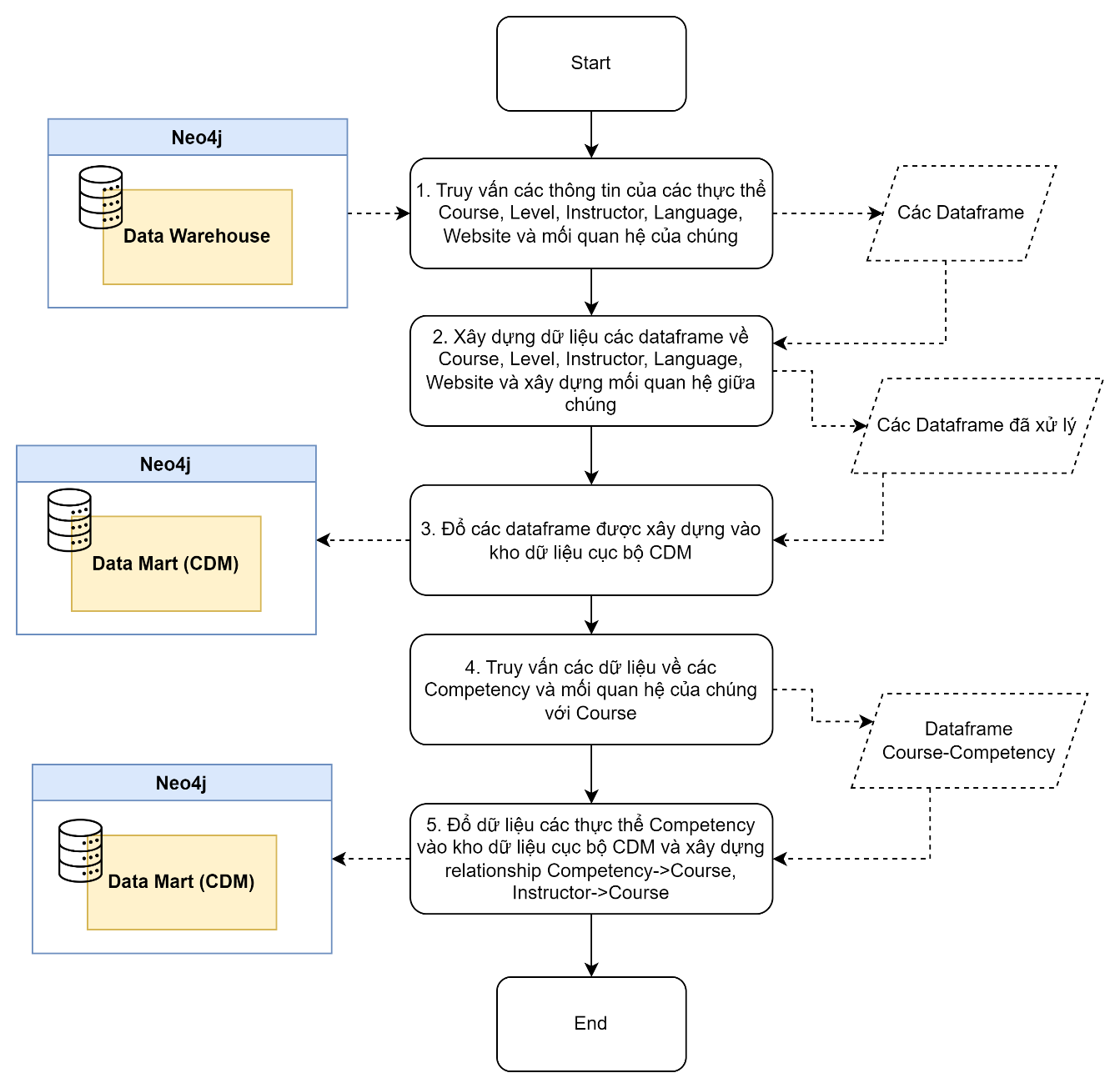
Diagram, text

Description automatically generated

Hình . Quá trình đổ dữ liệu từ GDW vào SDM

#### Sơ đồ chuyển dữ liệu từ GDW sang CDM

Với CDM (hình 6.7), các thông tin về Course và các thực thể liên quan như Level, Subtitle, Instructor, Website và mối quan hệ giữa chúng sẽ được giữ nguyên khi đưa vào CDM, nhưng mối quan hệ giữa Course với thực thể Instructor và với các thực thể Competency sẽ được đảo ngược chiều lại trước khi đưa vào CDM dựa trên nhu cầu phục vụ tìm kiếm được đưa ra ở YC4 ở mục 4.2.2 (chương 4) .



Hình . Quá trình đổ dữ liệu từ GDW vào CDM

#### Sơ đồ chuyển dữ liệu từ GDW sang UDM

UDM sẽ có quá trình xử lý đơn giản hơn (Hình 6.8), với UDM, quá trình xử lý chỉ cần truy vấn đồ thị con gồm các thực thể User và các thực thể Competency (bước 1), sau đó xây dựng các khung dữ liệu tương ứng (bước 2) và đổ trực tiếp vào UDM (bước 3).

A picture containing text

Description automatically generated

Hình . Quá trình đổ dữ liệu từ GDW vào UDM

#### Sơ đồ chuyển dữ liệu từ GDW sang CCDM

Cuối cùng, với CCDM, quá trình xử lý dữ liệu sẽ tìm đồ thị con chứa mối quan hệ giữa các khóa học có yêu cầu kỹ năng đầu vào và các khóa học có đào tạo những kỹ năng yêu cầu đấy, đồ thị con sau khi được Spark đọc từ GDW sẽ có khung dữ liệu gồm các trường gồm *course\_require\_id, competency, course\_teach\_id*, sau đó sẽ đổ vào CCDM và xây dựng mối quan hệ “TO” với khóa học yêu cầu kỹ năng sẽ nối với khóa học dạy kỹ năng đó dựa trên *course\_require\_id* và *course\_teach\_id* (hình 6.9).

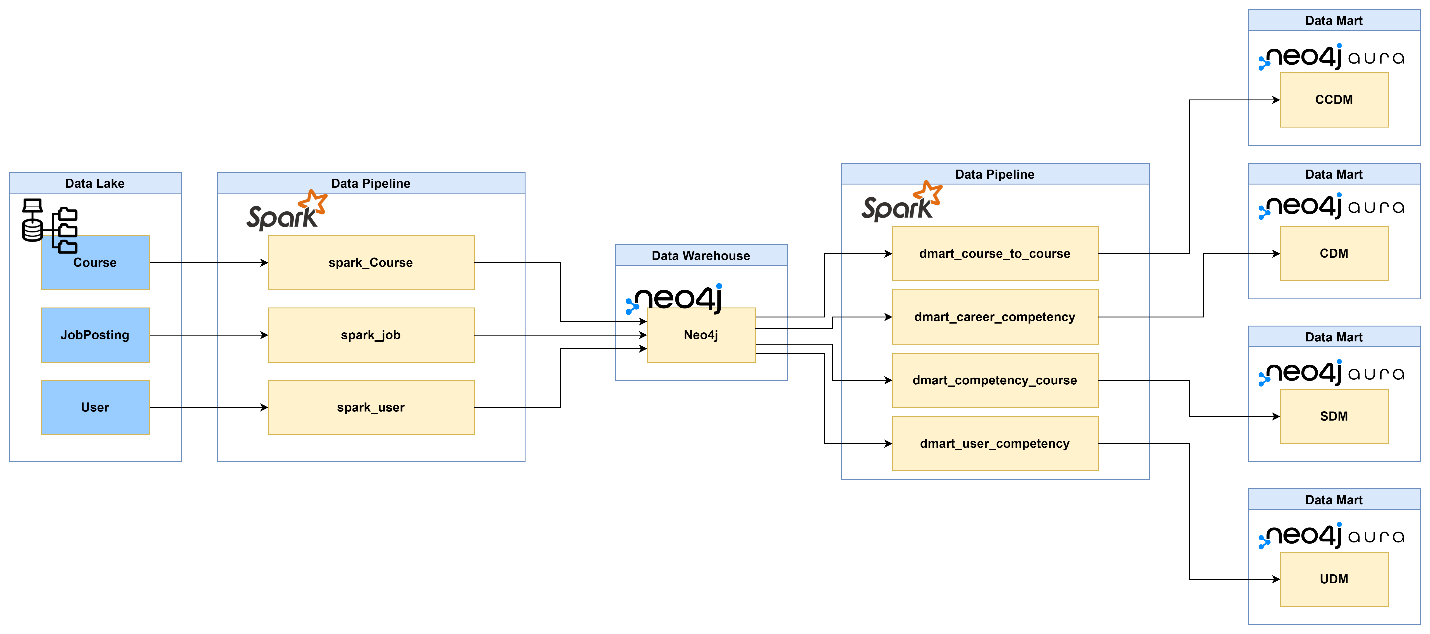
Diagram

Description automatically generated

Hình . Quá trình đổ dữ liệu từ GDW vào CCDM

### Kiến trúc đường ống dữ liệu

Sau khi đã xây dựng sơ đồ chi tiết quá trình xử lý dữ liệu cho từng bài toán, chúng em tiến hành xây dựng các đường ống dữ liệu (data pipeline) tương ứng dựa trên các quá trình đổ dữ liệu ở các mục 6.1.2, 6.1.3.1, 6.1.3.2, 6.1.3.3 và 6.1.3.4 để thực hiện quá trình ETL trong hệ thống GDW. Hình 6.10 thể hiện kiến trúc xây dựng các đường ống dữ liệu trong hệ thống từ kho dữ liệu thô sang GDW và từ GDW sang các GDM khác nhau.



Hình . Kiến trúc các đường ống dữ liệu trong hệ thống

Trong quá trình chạy, các đường ống *spark\_course, spark\_job, spark\_user* sẽ được chạy song song và đổ các dữ liệu xử lý vào GDW. Sau đó, các đường ống dữ liệu còn lại sẽ chạy song song, lấy các dữ liệu từ trong GDW và đổ vào các GDM. Trong đó, các đường ống dữ liệu *spark\_user, dmart\_user\_competency* sẽ được xử lý luồng, khi có tệp dữ liệu mới xuất hiện ở kho dữ liệu thô, tệp tin đó sẽ được xử lý ngay lập tức.

## Cài đặt hệ thống

### Thực hiện cài đặt

Chúng em sẽ tiến hành chạy các đường ống dữ liệu để xử lý các tệp tin của Course, Jobposting và User. Tổng khối lượng dữ liệu dùng để thử nghiệm là 4.5MB, với tổng số là 1468 bản ghi (*283 bản ghi về Course, 1182 bảng ghi về Jobposting, 3 bản ghi về User*). Sau khi chạy xong các đường ống dữ liệu xử lý các tệp tin, tiến hành chạy các đường ống dữ liệu để lấy dữ liệu từ GDW sang 4 GDM khác nhau. Các đường ống dữ liệu sẽ được chạy theo hình thức xử lý lô trong thử nghiệm này. Sau khi thử nghiệm xong hệ thống GDW, tiến hành cài đặt và thử nghiệm hệ thống kho dữ liệu truyền thống và sau đó so sánh kết quả thử nghiệm của hai hệ thống với số lượng dữ liệu lớn hơn và đưa ra kết luận.

Hệ thống thử nghiệm được triển khai tạm thời trên 1 nút máy (local). Máy này sẽ được xây dựng bằng máy ảo với phần mềm VMware Work Station 16, với Chi tiết cấu hình các máy ảo được liệt kê ở bảng 6.6.

Bảng . Cấu hình của máy ảo

|  |  |
| --- | --- |
| **Cấu hình** | **Thông tin cấu hình** |
| Vi xử lý | AMD Ryzen 5 3550H with Radeon Vega Mobile Gfx |
| Bộ nhớ trong | 12GB RAM |
| Bộ lưu trữ | 100GB SSD |
| Tổng lõi vi xử lý (processor core) được cung cấp cho máy ảo | 6 |
| Hệ điều hành | Ubuntu 20.04.5 LTS |

Hệ thống sẽ được đánh giá theo thời gian chạy khi xử lý dữ liệu và đổ dữ liệu vào GDW và các GDM. Nhóm đã thực hiện thử nghiệm đầu tiên với khối lượng dữ liệu là 4.5MB với 1468 bản ghi. Bảng 6.7 ghi lại chi tiết thời gian chạy của các đường ống dữ liệu và hình 6.10 thể hiện quá trình chạy của các đường ống dữ liệu trong hệ thống.

Bảng . Bảng chi tiết thời gian chạy của các đường ống dữ liệu

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Tên đường ống** | **Vai trò** | **Số bản ghi xử lý** | **Thời gian chạy (giây)** | **Số nút và quan hệ** |
| 1 | spark\_course | Đọc và xử lý dữ liệu khóa học từ hệ thống tập tin trong máy và đổ dữ liệu vào GDW. | 283 | 959,5 | 1220 nút (Course, Language, Website, Instructor, Competency) và 3390 quan hệ giữa chúng. |
| 2 | spark\_job | Đọc và xử lý dữ liệu các bài đăng tuyển dụng từ hệ thống tập tin trong máy và đổ dữ liệu vào GDW. | 1182 | 4200.3 | 2883 nút (Career, Jobposting, Competency) và 5584 mối quan hệ giữa chúng. |
| 3 | spark\_user | Đọc và xử lý dữ liệu người dùng từ hệ thống tập tin trong máy và đổ dữ liệu vào GDW. | 3 | 15.6 | 3 nút (User) và 27 quan hệ (Các quan hệ giữa user với Career và Competency). |
| 4 | dmart\_career\_LO | Xử lý và đổ dữ liệu từ GDW sang SDM. | X | 50.9 | 1884 nút (Career, Competency) và 3125 quan hệ giữa chúng. |
| 5 | dmart\_course\_course | Xử lý và đổ dữ liệu từ GDW sang CCDM. | X | 18.5 | 283 nút (Course) và 182 quan hệ giữa các Course. |
| 6 | dmart\_LO\_Course | Xử lý và đổ dữ liệu từ GDW sang CDM. | X | 63.8 | 1220 nút (Competency, Course, Instructor, Website, Language) và 3380 quan hệ giữa chúng. |
| 7 | dmart\_user\_LO | Xử lý và đổ dữ liệu từ GDW sang UDM. | X | 38.8 | 25 nút (User và Competency) và 27 quan hệ giữa chúng. |

### Kết quả cài đặt hệ thống

Chúng em thực hiện xây dựng các biểu đồ trực quan kết quả từ các GDM trong hệ thống để giám sát, theo dõi dữ liệu nhận được từ hệ thống. Các biểu đồ biểu diễn này được trực quan hóa bằng công cụ PowerBI, và kết quả trực quan thể hiện trong hình 6.11, 6.12, 6.13 và 6.14 sau.

Biểu đồ trực quan ở hình 6.11 biễu diễn dữ liệu về các khóa học đang được lưu trữ trong hệ thống. Các dữ liệu về khóa học bao gồm các thông tin chi tiết như tên khóa học, đường dẫn, học phí, đánh giá và thời gian giảng dạy của khóa học. Ngoài ra có các thông tin liên quan khác như ngôn ngữ giảng dạy, nền tảng cung cấp khóa học, người hướng dẫn và mức độ khó của khóa học. Bên cạnh đó, có thể thấy rõ sự phân bố về các kỹ năng mà những khóa học giảng dạy thông qua các biểu đồ cột ngang.

Graphical user interface

Description automatically generated

Hình . Biểu đồ trực quan dữ liệu khóa học trong CDM

Dữ liệu về nghiệp nghiệp được trực quan trong biểu đồ hình 6.12. Dữ liệu về nghề nghiệp bao gồm tên nghề nghiệp, số bài đăng tuyển dụng về nghề nghiệp đó và số bài đăng tuyển dụng nghề nghiệp có yêu cầu đối với từng kỹ năng. Từ biểu đồ có thể thấy xu hướng nghề nghiệp cũng như xu hướng về kỹ năng nghề nghiệp. Bên cạnh đó thông qua việc tương tác với bộ lọc nghề nghiệp trong biểu đồ, biểu đồ có thể thể hiện được các kỹ năng cần thiết trong mỗi nghề nghiệp.

Graphical user interface, chart

Description automatically generated

Hình . Biểu đồ trực quan dữ liệu về nghề nghiệp trong SDM

Dữ liệu về người dùng được trực quan hóa trong biểu đồ của hình 6.13. Trong biểu đồ biểu diễn các thông tin cơ bản của người dùng như tên, email và các kỹ năng mà người dùng đã có.

Graphical user interface, application

Description automatically generated

Hình . Biểu đồ trực quan dữ liệu người dùng trong UDM

Biểu đồ trực quan ở hình 6.14 biễu diễn trình tự của các khóa học theo kỹ năng đầu vào và đầu ra. Ví dụ với khóa học có id khóa học (Course ID) là 181, sẽ có những khóa học có id lần lượt là 66, 97, 140, 160, 173, 174, 187 đào tạo kỹ năng đầu vào của khóa học id 181.

Graphical user interface

Description automatically generated

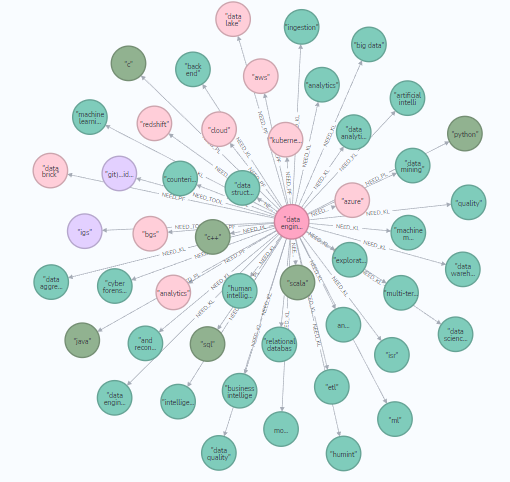
Hình . Biểu đồ trực quan trình tự của các khóa học trong CCDM

Từ dữ liệu trong các GDM đã được xây dựng, hệ thống có thể truy vấn và trả lời cho các yêu cầu của bài toán được nêu trong mục 3.2.

Với các yêu cầu trong nhóm phân tích thống kê bao gồm YC1, YC2 và YC3 trong bảng 4.1, SDM được sử dụng để truy vấn dữ liệu cần thiết. Chúng em minh họa cho các yêu cầu này thông qua một trường hợp cụ thể là truy vấn các thông tin liên quan đến nghề nghiệp “data engineer”. Với YC1 là truy vấn số lượng bài đăng tuyển dụng nghề nghiệp “data engineer. Tương ứng với YC2 là với nghề nghiệp “data engineer”, tính số lượng bài đăng có yêu cầu theo mỗi kỹ năng .Và trong YC3 là tính số lượng bài đăng tuyển dụng “data engineer” có yêu cầu kỹ năng “python”. Trong hình 6.16 và 6.17 là kết quả thực thi câu truy vấn trong hình 6.15 để trả lời cho các yêu cầu trong ví dụ minh họa trên. Hình 6.18 là số lượng bài đăng được đăng tuyển nghề nghiệp “data engineer” và hình 6.19 là số lượng bài đăng yêu cầu kỹ năng “python” cho nghề nghiệp này.



Hình . Câu truy vấn sử dụng ở SDM



Hình . Kết quả truy vấn phục vụ bài toán ở SDM

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình . Kết quả truy vấn với các thông tin về đồ thị ở SDM

Graphical user interface, text, application

Description automatically generated

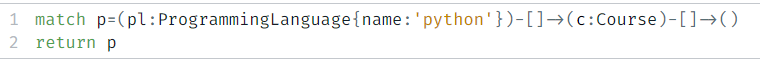
Hình . Số lượng bài đăng của nghề nghiệp “data engineer”

Graphical user interface, text, application, chat or text message

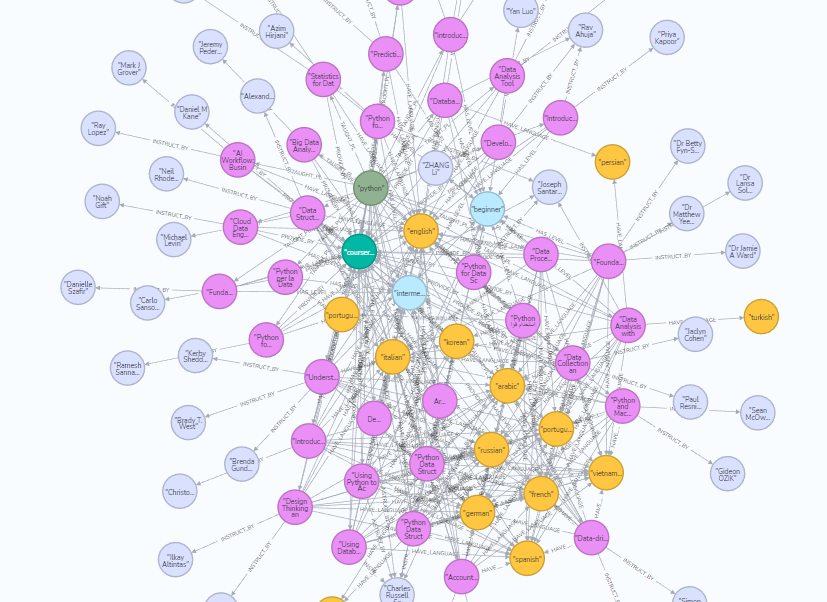
Description automatically generated

Hình . Số lượng bài đăng yêu cầu kỹ năng “python” cho nghề nghiệp “data engineer”

Với YC4 trong nhóm chức năng tìm kiếm trong bảng 4.3, yêu cầu xây dựng CDM để phục vụ tìm kiếm khóa học theo kỹ năng nghề nghiệp. Hình 6.20 là câu truy vấn sử dụng và hình 6.21, 6.22 là kết quả truy vấn các khóa học có hướng dẫn ngôn ngữ lập trình “python”, với các thông tin của khóa học bao gồm mức độ khó của khóa học, người hướng dẫn và ngôn ngữ của các khóa học đó.



Hình . Câu truy vấn sử dụng ở CDM



Hình . Kết quả truy vấn các khóa học có hướng dẫn ngôn ngữ lập trình “python” ở CDM

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình . Kết quả truy vấn với các thông tin về đồ thị CDM

Tiếp theo, kiểm tra thử câu truy vấn cho UDM, để phục vụ cho YC5 và YC6 trong bảng 4.5, chúng em thực hiện câu truy vấn tìm các kỹ năng mà một người dùng đã có. Hình 6.23 là câu truy vấn sử dụng và 6.24, 6.25 là kết quả truy vấn người dùng có *username* là “toan” trong UDM và các kỹ năng mà người dùng đã có.

A picture containing text

Description automatically generated

Hình . Câu truy vấn được sử dụng ở UDM

Diagram

Description automatically generated

Hình . Kết quả truy vấn thông tin của một người dùng có username là “toan” ở UDM

Graphical user interface, text, application, chat or text message

Description automatically generated

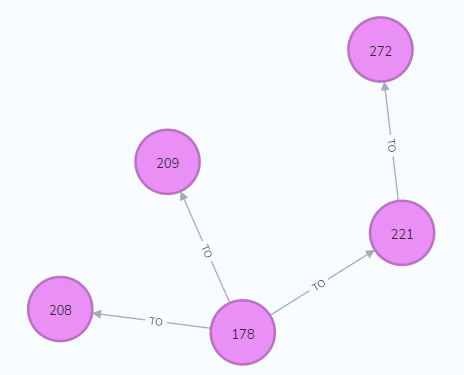
Hình . Kết quả truy vấn với các thông tin về đồ thị UDM

Cuối cùng, hình 6.27, 6.28 là kết quả truy vấn ở CCDM cho một khóa học với đầu vào là *id* của khóa học đó, kết quả là một đồ thị con tập hợp các khóa học có thể học để đạt mục tiêu nghề nghiệp dựa trên khóa học đầu vào. Hình 6.26 là câu truy vấn được sử dụng ở CCDM.

Text

Description automatically generated with medium confidence

Hình . Câu truy vấn sử dụng ở CCDM



Hình . Kết quả truy vấn cho tập hợp các khóa học cần học cho khóa học có id là 178 ở CCDM

Graphical user interface, text, application, chat or text message

Description automatically generated

Hình . Kết quả với các thông tin về đồ thị CCDM

## Thử nghiệm và phân tích kết quả

### Kịch bản thử nghiệm

Chúng em thực hiện xây dựng thêm một hệ thống TDW tương đương với hệ thống GDW để thực hiện đối sánh dựa trên các tiêu chí theo tiêu chuẩn chủ quan (tốc độ đọc dựa vào các câu truy vấn, khả năng mở rộng), tiêu chuẩn chủ quan (khả năng hỗ trợ, linh hoạt và dễ cài đặt hay không). Cơ sở dữ liệu áp dụng cho hệ thống TDW sẽ là SQL Server 2019 - một cơ sở dữ liệu quan hệ được phát triển bởi Microsoft. SQL Server 2019 sẽ được cài đặt bằng Docker. Đồng thời, để kết quả thực nghiệm có thể chính xác hơn thì chúng em cũng cài đặt Neo4j ở máy cục bộ bằng Docker để kết quả đối sánh có thể chính xác hơn. Hình 6.29 là lược đồ kho dữ liệu quan hệ (bông tuyết) được áp dụng trong hệ thống TDW để thực hiện thử nghiệm, dựa trên kiến trúc đồ thị được đưa ra ở hình 3.3. Bảng 6.8 là mô tả chi tiết cho các bảng trong lược đồ TDW.

**Diagram

Description automatically generated**

Hình . Lược đồ kho dữ liệu quan hệ thử nghiệm

Bảng . Bảng mô tả chi tiết các bảng và ý nghĩa của chúng trong TDW

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên bảng** | **Ý nghĩa** |
| 1 | dim\_instructor | Bảng chiều lưu trữ dữ liệu về người hướng dẫn khóa học. |
| 2 | dim\_language | Bảng chiều lưu trữ dữ liệu về các ngôn ngữ/phụ đề. |
| 3 | dim\_level | Bảng chiều lưu trữ dữ liệu về các mức độ khó. |
| 4 | dim\_website | Bảng chiều lưu trữ dữ liệu về các nền tảng E-learning. |
| 5 | dim\_course | Bảng chiều lưu trữ dữ liệu về các khóa học. Chứa các thông tin về khóa học và liên kết với các thông tin khác như dim\_level và dim\_website bằng khóa ngoại và dim\_language và dim\_instructor bằng bảng cầu nối (brigde table). |
| 6 | brige\_instructor\_course | Bảng cầu nối giữa bảng chiều dim\_instructor và bảng chiều dim\_course, lưu trữ các dữ liệu của mối quan hệ N:N giữa 2 thực thể Instructor và Course. |
| 7 | bridge\_language\_course | Bảng cầu nối giữa bảng chiều dim\_instructor và bảng chiều dim\_course, lưu trữ các dữ liệu của mối quan hệ N:N giữa 2 thực thể Language và Course. |
| 8 | fact\_course | Bảng sự kiện về khóa học, chứa các thông tin chi tiết của 1 khóa học gồm khóa chính của khóa học (courseKey) để biết đó là dữ kiện của khóa học nào, khóa học đó dạy kỹ năng nghề nghiệp nào (competencyId), và kỹ năng nghề nghiệp này là được dạy hay đó là kỹ năng yêu cầu đầu vào (is\_require). |
| 9 | dim\_competency | Bảng chiều lưu trữ dữ liệu về các kỹ năng nghề nghiệp CNTT. |
| 10 | dim\_career | Bảng chiều lưu trữ dữ liệu về các ngành nghề CNTT. |
| 11 | fact\_jobposting | Bảng sự kiện về các bài đăng tuyển dụng, chứa các thông tin chi tiết của 1 bài đăng tuyển dụng bao gồm bài đăng tuyển dụng đó cần tuyển nghề nghiệp nào (careerId), các kỹ năng nghề nghiệp bài bài đăng đó yêu cầu (competencyId) và ngày đăng bài đăng tuyển dụng (jobPostDate). |
| 12 | dim\_user | Bảng chiều lưu trữ dữ liệu về người dùng. |
| 13 | fact\_user | Bảng sự kiện về các thông tin về người dùng, chứa thông tin về các kỹ năng nghề nghiệp mà người dùng đã có (competencyId), người dùng thích ngành nghề nào (careerId). |

Sau khi xây dựng xong hệ thống TDW, chúng em sử dụng CDM của hai hệ thống kho dữ liệu đồ thị và kho dữ liệu truyền thống để thử nghiệm tốc độ truy vấn dữ liệu. Bộ dữ liệu thử nghiệm là các dữ liệu khóa học sau khi được thực hiện rút trích thực thể sau đó đổ vào trong hai hệ thống, với tập dữ liệu thô có cấu trúc tương tự như bảng 6.3 nhưng với khối lượng dữ liệu khác nhau. Trong thử nghiệm lần này, chúng em thực hiện thử nghiệm truy vấn không phân cấp và truy vấn có phân cấp để thực hiện đối sánh giữa hai hệ thống. Do trong lược đồ kho dữ liệu truyền thống có thêm bảng cầu nối để lưu trữ mối quan hệ nhiều-nhiều giữa bảng chiều dim\_course với hai bảng chiều dim\_instructor và dim language nên mức độ phân cấp tối đa của TDW là 3. Trong khi đó do lược đồ GDW liên kết trực tiếp giữa Course với Instructor và Language nên mức độ phân cấp tối đa của GDW là 2. Bảng 6.9 là bảng chi tiết các câu truy vấn sẽ thử nghiệm cho hai hệ thống và các bảng cần kết trong TDW và các thực thể, quan hệ cần thực hiện phép duyệt đồ thị trong GDW.

Bảng . Bảng chi tiết các trường hợp phân cấp và các bảng/nút và quan hệ trong câu truy vấn của hai hệ thống

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Loại truy vấn** | **Câu truy vấn** | **Loại hệ thống** | **Bảng hoặc nút/quan hệ** | **Ý nghĩa câu truy vấn thực nghiệm** |
| Truy vấn không phân cấp | Q1 | TDW | dim\_course | Truy vấn thông tin cơ bản của các khóa học có lượt đánh giá > 4.8. |
| GDW | Nút: Course |
| Q2 | TDW | dim\_course, dim\_level. | Truy vấn thông tin các khóa học có lượt đánh giá > 4.8, bao gồm thông tin cơ bản và mức độ khó của khóa học |
| GDW | Nút: Course, Level.  Quan hệ: HAS\_LEVEL. |
| Truy vấn phân cấp | Q3 | TDW | fact\_course, dim\_course, dim\_competency, dim\_website, dim\_level. | Truy vấn các thông tin về khóa học bao gồm cả mức độ khó và nền tảng của khóa học đó. |
| GDW | Nút: Course, ProgrammingLanguage, Website, Level.  Quan hệ: TAUGHT\_PL, PROVIDE\_BY, HAS\_LEVEL. |
| Q4 | TDW | fact\_course, dim\_course, dim\_competency, dim\_website, dim\_level, brigde\_instructor\_course, brigde\_language\_course,dim\_instructor, dim\_language | Truy vấn các thông tin về khóa học bao gồm cả mức độ khó, nền tảng, người hướng dẫn và ngôn ngữ của khóa học đó |
| GDW | Nút: Course, ProgrammingLanguage, Website, Level, Instructor, Language.  Quan hệ: TAUGHT\_PL, PROVIDE\_BY, HAS\_LEVEL, INSTRUCT, HAVE\_LANGUAGE. |

#### Trường hợp 1: Truy vấn không phân cấp.

Ở trường hợp 1, chúng em thực hiện thử nghiệm 2 câu truy vấn không phân cấp là Q1 và Q2. Hình 6.30, 6.31 là câu truy vấn Q1 áp dụng cho 2 hệ thống. Bảng 6.10 là kết quả thời gian truy vấn không phân cấp Q1 từ CDM của hai hệ thống với khối lượng dữ liệu tương tự và hình 6.32 là biểu đồ trực quan cho kết quả này.

Graphical user interface, text

Description automatically generated

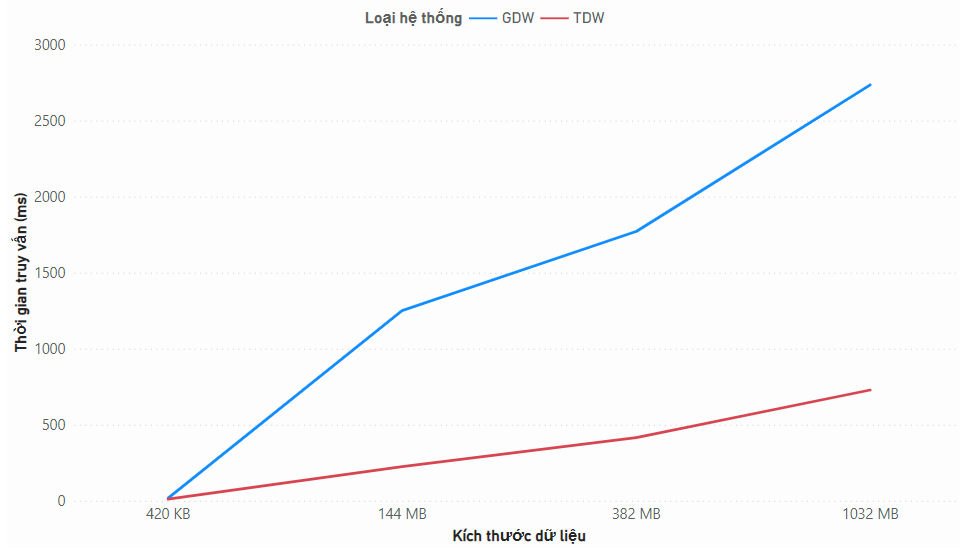
Hình . Câu truy vấn không phân cấp Q1 sử dụng SQL



Hình . Câu truy vấn không phân cấp Q1 sử dụng Cypher

Bảng . Bảng kết quả thời gian truy vấn không phân cấp Q1

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Khối lượng dữ liệu** | **Loại hệ thống** | **Thời gian truy vấn** |
| 1 | 420kb | GDW | 18 ms |
| TDW | 11 ms |
| 2 | 144mb | GDW | 1251 ms |
| TDW | 225 ms |
| 3 | 382mb | GDW | 1771 ms |
| TDW | 416 ms |
| 4 | 1032mb | GDW | 2735 ms |
| TDW | 729 ms |



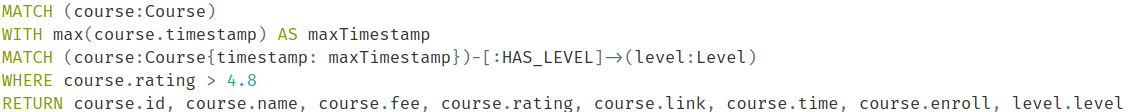
Hình . Biểu đồ trực quan kết quả thời gian truy vấn không phân cấp Q1 của hai hệ thống

Tiếp tục, chúng em thực hiện thử nghiệm truy vấn không phân cấp Q2. Hình 6.33, 6.34 là hai câu truy vấn Q2 và áp dụng cho 2 hệ thống. Bảng 6.11 là kết quả thời gian truy vấn không phân cấp từ CDM của hai hệ thống với khối lượng dữ liệu tương tự và hình 6.35 là biểu đồ trực quan cho kết quả này.

Text

Description automatically generated

Hình . Câu truy vấn không phân cấp Q2 sử dụng SQL



Hình . Câu truy vấn không phân cấp Q2 sử dụng Cypher

Bảng . Bảng kết quả thời gian truy vấn không phân cấp Q2 của hai hệ thống với khối lượng dữ liệu khác nhau

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Khối lượng dữ liệu** | **Loại hệ thống** | **Thời gian truy vấn** |
| 1 | 420kb | GDW | 12 ms |
| TDW | 10 ms |
| 2 | 144mb | GDW | 1949 ms |
| TDW | 217 ms |
| 3 | 382mb | GDW | 2581 ms |
| TDW | 339 ms |
| 4 | 1032mb | GDW | 4037 ms |
| TDW | 866ms |

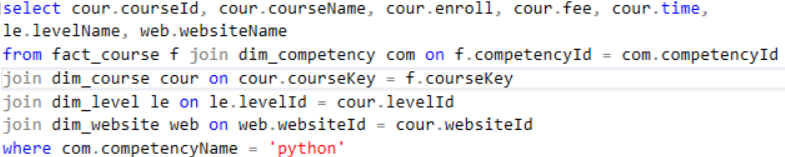
Chart, line chart

Description automatically generated

Hình . Biểu đồ trực quan kết quả thời gian truy vấn không phân cấp Q2 của hai hệ thống

#### Trường hợp 2: Truy vấn có phân cấp.

Ở trường hợp 2, chúng em thử nghiệm truy vấn với mức độ phân cấp 2 với câu truy vấn Q3. Hình 6.36 là câu truy vấn Q3 có phân cấp (mức độ phân cấp 2) áp dụng để thử nghiệm cho TDW và hình 6.37 là câu truy vấn Q3 tương tự áp dụng cho GDW. Bảng 6.12 là bảng kết quả thời gian truy vấn có phân cấp từ CDM của hai hệ thống với khối lượng dữ liệu khác nhau theo 3 mức từ nhỏ tới lớn. Hình 6.38 là biểu đồ trực quan cho kết quả này.



Hình . Câu truy vấn có phân cấp Q3 (mức độ phân cấp 2) sử dụng SQL

Text

Description automatically generated

Hình . Câu truy vấn có phân cấp Q3 (mức độ phân cấp 2) sử dụng Cypher

Bảng . Bảng kết quả thời gian truy vấn phân cấp Q3 (mức độ phân cấp 2) của hai hệ thống với khối lượng dữ liệu khác nhau

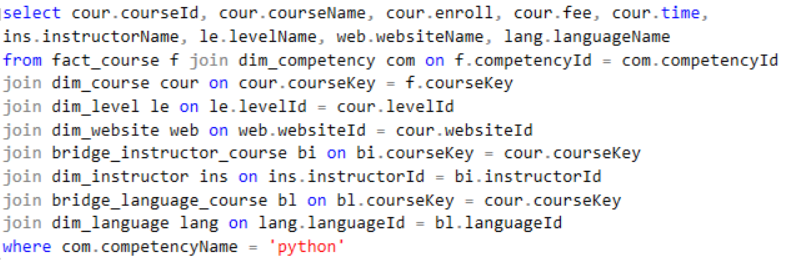
|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Khối lượng dữ liệu** | **Loại hệ thống** | **Thời gian truy vấn** |
| 1 | 420kb | GDW | 2 ms |
| TDW | 3 ms |
| 2 | 144mb | GDW | 17 ms |
| TDW | 115 ms |
| 3 | 382mb | GDW | 42 ms |
| TDW | 177 ms |
| 4 | 1032mb | GDW | 86 ms |
| TDW | 562 ms |

Chart, line chart

Description automatically generated

Hình . Biểu đồ trực quan kết quả thời gian truy vấn phân cấp Q3 (mức độ phân cấp 2) của hai hệ thống

Tiếp tục, chúng em tiếp tục thử nghiệm truy vấn có phân cấp với Q4. Với cùng một yêu cầu nhưng câu truy vấn ở GDW sẽ có mức độ phân cấp là 2 còn ở TDW sẽ có mức độ phân cấp là 3. Hình 6.39 là câu truy vấn Q4 có phân cấp (mức độ phân cấp 3) áp dụng để thử nghiệm cho TDW và hình 6.40 là câu truy vấn Q4 có mức độ phân cấp 2 áp dụng cho GDW. Bảng 6.13 là bảng kết quả thời gian truy vấn có phân cấp từ CDM của hai hệ thống với khối lượng dữ liệu khác nhau. Hình 6.41 là biểu đồ trực quan cho kết quả này.



Hình . Câu truy vấn có phân cấp Q4 (mức độ phân cấp 3) sử dụng SQL

Text

Description automatically generated

Hình . Câu truy vấn có phân cấp Q4 (mức độ phân cấp 2) sử dụng Cypher

Bảng . Bảng kết quả thời gian truy vấn có phân cấp Q4 (mức độ phân cấp 3-2) của hai hệ thống với khối lượng dữ liệu khác nhau

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Khối lượng dữ liệu** | **Loại hệ thống** | **Thời gian truy vấn** |
| 1 | 420kb | GDW | 3 ms |
| TDW | 5 ms |
| 2 | 144mb | GDW | 24 ms |
| TDW | 223 ms |
| 3 | 382mb | GDW | 33 ms |
| TDW | 546 ms |
| 4 | 1032mb | GDW | 67 ms |
| TDW | 1454 ms |

Chart, line chart

Description automatically generated

Hình . Biểu đồ trực quan kết quả thời gian truy vấn phân cấp của hai hệ thống

Từ kết quả của các thử nghiệm trên, ta thấy được tốc độ truy vấn của hệ thống TDW nhanh hơn so với hệ thống GDW khi truy vấn không phân cấp do TDW không cần thực hiện phép kết nhiều bảng, trong khi GDW cần phải duyệt đồ thị để tìm kiếm các thực thể trong câu truy vấn để trả về kết quả (câu truy vấn Q2). Trong trường hợp chỉ truy vấn thông tin về 1 khóa học (câu truy vấn Q1), GDW cần duyệt tất cả nút Course và tìm lấy ra thuộc tính *rating* của nút, sau đó lọc theo điều kiện câu truy vấn mới trả về kết quả, nên thời gian truy vấn của GDW ở trường hợp này chậm hơn so với TDW chỉ truy vấn dữ liệu ở một cột trong một bảng.

Tuy nhiên với truy vấn có phân cấp (các câu truy vấn Q3 và Q4) thì GDW có tốc độ truy vấn nhanh hơn TDW. Đặc biệt khi mức độ phân cấp càng nhiều thì sự chênh lệch thời gian truy vấn giữa TDW và GDW càng cao, sự khác biệt về thời gian truy vấn càng rõ khi kích thước dữ liệu càng lớn. Nhờ vào khả năng duyệt đồ thị mạnh mẽ của cơ sở dữ liệu đồ thị so với phép toán kết nhiều bảng giữa bảng dữ kiện và các bảng chiều trong phân cấp của hệ thống TDW. Đồng thời, do nhu cầu tìm kiếm khóa học với đầu vào là một kỹ năng nghề nghiệp (như các câu truy vấn trên đều tìm kiếm kỹ năng “python”), sau đó truy vấn các thông tin về khóa học có dạy kỹ năng đó, nên GDW chỉ cần tìm kiếm nút ProgrammingLanguage và không cần duyệt các loại nút kỹ năng nghề nghiệp khác thay vì duyệt toàn bộ bảng chứa tất cả các loại kỹ năng trong một bảng chiều dim\_competency như trong TDW, do đó tối ưu hóa được thời gian truy vấn của GDW. Từ đó thấy được ưu điểm của kho dữ liệu đồ thị khi về mặt lược đồ, kho dữ liệu đồ thị có thể lưu trữ mối quan hệ nhiều-nhiều một cách dễ dàng, trong khi TDW cần phải tạo các bảng cầu nối để có thể lưu trữ các mối quan hệ Course với Instructor và Language. Đồng thời với lược đồ đồ thị, các thực thể trong lớp Competency được chia làm 5 loại khác nhau (ProgrammingLanguage, Tool, Framework, Platform, Knowledge), thay vì lưu trữ hết toàn bộ các loại kỹ năng vào 1 thực thể như bảng chiều dim\_competency như TDW, từ đó có thể tối ưu hóa được câu truy vấn của GDW, không cần truy vấn hết tất cả các thực thể kỹ năng trong lớp Competency. Về mặt truy vấn, tốc độ truy vấn khi càng nhiều phân cấp thì tốc độ truy vấn của GDW càng áp đảo hơn so với TDW.

### Đánh giá kết quả

Hệ thống GDW thử nghiệm đã được cài đặt thành công trên 1 nút máy ảo để thực hiện các công việc xử lý và lưu trữ dữ liệu để phục vụ cho các yêu cầu bài toán của đề tài. Có thể áp dụng mở rộng hệ thống này bằng cách xây dựng một hệ thống hoàn chỉnh hơn với hệ thống kho dữ liệu thô phân tán của Hadoop (Hadoop File System, thường gọi là HDFS) để lưu trữ các tập tin dữ liệu thô, thêm tài nguyên cho máy và xây dựng hệ thống phân tán để có thể tối ưu hóa khả năng xử lý dữ liệu của hệ thống. Sau đây là bảng 6.14 đánh giá mức độ hoàn thiện của hệ thống thử nghiệm.

Bảng . Đánh giá mức độ hoàn thiện các chức năng trong hệ thống

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mức độ hoàn thiện** |
| 1 | Các đường ống dữ liệu xử lý dữ liệu dữ liệu thô và đổ vào GDW. | 100% |
| 2 | Các đường ống dữ liệu chuyển dữ liệu từ GDW sang các SDM, CDM, UDM, CCDM. | 100% |
| 3 | Cài đặt TDW và đối sánh với GDW khi truy vấn phân cấp và truy vấn không phân cấp. | 100% |

Với kết quả thời gian chạy và quá trình thực hiện xây dựng hệ thống thử nghiệm thì chúng em có những đánh giá sau:

* Tiêu chuẩn khách quan:
  + Về tốc độ truy vấn:
    - Đối với các câu truy vấn phân cấp, GDW có tốc độ truy vấn nhanh hơn nhiều so với TDW. Khi mức độ phân cấp càng nhiều và dữ liệu càng lớn thì thời gian truy vấn của GDW càng áp đảo hơn so với TDW. Do TDW cần phải thực hiện phép toán kết bảng còn GDW thì chỉ cần thực hiện phép duyệt đồ thị. Điều này càng được thấy rõ hơn khi khối lượng dữ liệu càng lớn.
    - Đối với câu truy vấn không phân cấp, kho dữ liệu quan hệ có tốc độ truy vấn nhanh hơn so với kho dữ liệu đồ thị do TDW câu truy vấn không quá phức tạp, không cần phải thực hiện quá nhiều pháp kết bảng, trong khi GDW cần phải duyệt đồ thị để tìm kiếm các thực thể trong câu truy vấn và thuộc tính để trả về kết quả. vì vậy, thời gian duyệt đồ thị trong kho dữ liệu đồ thị sẽ chậm hơn việc truy vấn dữ liệu trên một bảng của kho dữ liệu quan hệ.
  + Về tốc độ ghi dữ liệu (spark): Khi sử dụng Spark để đổ dữ liệu đã xử lý vào hệ thống thì GDW có tốc độ ghi dữ liệu chậm hơn rất nhiều so với TDW do trong quá trình đổ dữ liệu vào GDW, Neo4j cần tạo mối quan hệ giữa các nút. Trong quá trình nghiên cứu cài đặt, khi đổ toàn bộ dữ liệu đã xử lý liên quan tới bài đăng tuyển dụng thì thời gian đổ dữ liệu của GDW tới 4200.2 giây (với 1182 bản ghi), trong khi TDW có thời gian đổ dữ liệu chỉ mất 462 giây với số bản ghi tương ứng, ứng nghiệm với công trình nghiên cứu [4], tốc độ ghi dữ liệu của GDW sẽ chậm hơn TDW gần 14 lần.
  + Về khả năng mở rộng: Cả GDW và TDW đều có thể mở rộng theo chiều dọc (Vertical scaling) bằng cách thêm tài nguyên cho máy tính như CPU, bộ nhớ và dung lượng lưu trữ cho máy. Đồng thời CSDL của hai hệ thống này đều có thể hỗ trợ sử dụng các kỹ thuật như nhân bản (Replication), phân vùng dữ liệu (Sharding) và phân tán (Distrubuted) để mở rộng theo chiều ngang (Horizontal Scaling)
* Tiêu chuẩn chủ quan:
  + Về khả năng hỗ trợ: Các cơ sở dữ liệu để xây dựng TDW và GDW (Neo4j) đều có tài liệu và cộng đồng lớn mạnh do đã có thời gian phát triển đã khá lâu. Tuy nhiên, về mặt tài liệu và cộng đồng xây dựng TDW và GDW thì TDW có cộng đồng lớn mạnh hơn và được áp dụng rộng rãi ở giới công nghiệp. Còn GDW thì chỉ trong thời gian gần đây được nghiên cứu và bắt đầu áp dụng nên cộng đồng chưa được lớn mạnh.
  + Về độ linh hoạt: GDW có khả năng linh hoạt tốt hơn trong tương lai vì khi có sự thay đổi trong kiến trúc đồ thị như thêm thực thể mới hay thay đổi kiểu dữ liệu thì GDW không cần thay đổi kiến trúc tổng thể. Trong khi đó, TDW mở rộng bằng cách thêm bảng hoặc cột mới vào CSDL, hoặc khi cần lưu trữ các loại dữ liệu mới thì người dùng cần phải thêm bảng mới hoặc mở rộng cột của bảng hiện có, điều này khiến cho việc mở rộng của TDW sẽ tăng độ phức tạp của CSDL, đặc biệt khi số lượng bảng và cột tăng lên. Trong khi đó GDW chỉ cần thêm các nút mới hoặc cạnh mới vào đồ thị và có thể dễ dàng đáp ứng các yêu cầu mới trong tương lai.
  + Về độ khó cài đặt: TDW có độ khó cài đặt cao hơn khi cần ứng dụng các kỹ thuật như chiều thay đổi chậm (Slowly Change Dimension), hoặc kỹ thuật tạo các bảng cầu nối (bridge table) để có thể lưu trữ các mối quan hệ nhiều-nhiều trong lược đồ kho dữ liệu quan hệ như quan hệ nhiều-nhiều giữa Instructor và Language với Course. Trong khi GDW không cần các kỹ thuật quá phức tạp và sử dụng kỹ thuật đơn giản hơn để xử lý dữ liệu.

Từ các kết quả đánh giá trên cho thấy được GDW có ưu thế hơn TDW khi ứng dụng vào xây dựng bài toán của đề tài. Bởi dù TDW có ưu thế mạnh hơn so với GDW về tốc độ ghi dữ liệu, tốc độ truy vấn không phân cấp, cùng với tài liệu và cộng động lớn mạnh hơn nhưng các ưu thế nổi trội của GDW về tốc độ đọc dữ liệu ở trường hợp truy vấn có phân cấp, độ linh hoạt và dễ cài đặt hơn so với TDW khiến cho GDW là một giải pháp hợp lý cho bài toán.

## Kết luận chương

*Ở chương này, chúng em trình bày chi tiết về quá trình thử nghiệm hệ thống GDW bao gồm các công nghệ triển khai và các vai trò của chúng trong hệ thống. Bên cạnh đó cũng đưa ra chi tiết các quá trình xử lý và đổ dữ liệu từ kho dữ liệu thô vào GDW và chuyển dữ liệu từ GDW sang GDM, cuối cùng đưa ra kết quả chạy của hệ thống sau khi thử nghiệm.*

*Đồng thời, chúng em cũng xây dựng hệ thống TDW để thực hiện đối sánh với GDW, nhóm đã dựa trên kết quả thử nghiệm và quá trình thực nghiệm giữa hai hệ thống đã đưa ra kết luận GDW có ưu điểm hơn so với TDW trong tốc độ đọc dữ liệu và khả năng mở rộng cũng như tính linh hoạt và độ khó cài đặt.*

**Chương 7**

# Kết luận và hướng phát triển

*Chương này sẽ trình bày về những kết quả chúng em đã thực hiện được trong đề tài, tổng hợp những kiến thức mà chúng em đã học được trong suốt quá trình thực hiện luận văn. Đồng thời, đưa ra các đánh giá, nhận xét về những kết qủa đó và cuối cùng là đưa ra hướng phát triển của đề tài trong tương lai.*

## Kết luận

Sau quá trình phân tích và nghiên cứu về phương pháp xây dựng một kho dữ liệu đồ thị dựa trên nền tảng dữ liệu lớn cho đề xuất khóa học dựa trên mục tiêu nghề nghiệp CNTT, cụ thể là cung cấp dữ liệu cho ứng dụng thống kê xu hướng nghề nghiệp, kỹ năng nghề nghiệp và tư vấn khóa học cho mục tiêu nghề nghiệp trong lĩnh vực CNTT, chúng em đã học thêm được rất nhiều kiến thức từ đó hoàn thành bài toán đặt ra ở đề tài. Trong quá trình tìm hiểu và phát triển đề tài, sự sai sót và vướng mắc là điều không thể tránh khỏi, nhưng nhờ sự chỉ dẫn và giúp đỡ từ giáo viên hướng dẫn đã giúp chúng em điều chỉnh và hoàn thành bài nghiên cứu. Chúng em sẽ trình bày chi tiết các kết quả đã đạt được về mặt lý thuyết và triển khai hệ thống, bên cạnh đó cũng đưa ra các hạn chế liên quan.

### Kết quả đạt được về mặt lý thuyết

* Tìm hiểu về các kiến thức về kho dữ liệu, kho dữ liệu theo nhóm chức năng.
* Tìm hiểu về mô hình đồ thị, các công trình liên quan từ đó áp dụng mô hình đồ thị vào hệ thống kho dữ liệu.
* Đưa ra bảng so sánh kiến trúc giữa hệ thống kho dữ liệu đồ thị và hệ thống kho dữ liệu quan hệ truyền thống.
* Tìm hiểu về bài toán đề xuất khóa học dựa trên mục tiêu nghề nghiệp CNTT, từ đó phân tích các yêu cầu cụ thể của bài toán.
* Tìm hiểu cách thiết kế kiến trúc đồ thị, thiết kế kiến trúc của hệ thống và thiết kế mô hình GDM cho các nhóm chức năng trong yêu cầu của bài toán.
* Tìm hiểu về các công nghệ hỗ trợ cho bài toán, bao gồm công nghệ xử lý dữ liệu lớn và công nghệ lưu trữ CSDL đồ thị, từ đó lựa chọn công cụ phù hợp cho bước triển khai hệ thống.

### Kết quả đạt được về mặt triển khai hệ thống

Trong khóa luận này, chúng em đã thành công triển khai hệ thống cho bài toán của đề tài bao gồm:

* Xây dựng thành công hệ thống kho dữ liệu đồ thị dựa trên nền tảng dự liệu lớn cho đề xuất khóa học dựa trên mục tiêu nghề nghiệp CNTT.
* Xây dựng thêm hệ thống kho dữ liệu truyền thống để thực hiện đối sánh đưa ra các ưu điểm khác nhau giữa kho dữ liệu truyền thống và kho dữ liệu đồ thị.

Từ kết quả triển khai hệ thống, chúng em lập bảng đối sánh và chứng minh được phương pháp áp dụng mô hình đồ thị cho kiến trúc của hệ thống là hợp lý hơn so với áp dụng mô hình quan hệ truyền thống.

### Các hạn chế

Vì mặt hạn chế về tài nguyên nên hệ thống chỉ được cài đặt và xử lý dữ liệu trên một thiết bị, chưa thể áp dụng xử lý phân tán trên cụm máy dẫn đến tốc độ xử lý và đổ dữ liệu vào kho dữ liệu chưa được tối ưu.

Bên cạnh đó, các tiêu đề trong các bài đăng tuyển dụng khá phức tạp, lộn xộn nên chưa thể xử lý hoàn hảo được. Nhiều kỹ năng nghề nghiệp được nhận diện bởi NER không được chính xác.

## Hướng phát triển

Từ kết quả triển khai và những hạn chế mà nhóm đã thực hiện và gặp phải trong khóa luận này, nhóm đề xuất một số hướng phát triển thêm cho đề tài để đề tài hoàn thiện hơn:

* Áp dụng các công cụ và nền tảng dữ liệu lớn để xây dựng kho dữ liệu thô cho hệ thống.
* Tối ưu hóa quá trình xử lý dữ liệu trong các đường ống dữ liệu, tối ưu hóa và cải tiến mô hình NER cho hệ thống. Xây dựng thêm một mô hình học máy dự đoán nghề nghiệp dựa trên tiêu đề trong các bài đăng tuyển dụng.
* Mở rộng thêm tài nguyên cho hệ thống và áp dụng xử lý phân tán trên cụm máy với nhiều nút máy thực thi.

# Tài liệu tham khảo

|  |  |
| --- | --- |
| [1] | R. Kimball, "Data warehouse là gì? những tính năng bạn cần nên biết," [Online]. Available: https://ralphkimball.com/data-warehouse-la-gi/. [Accessed 9 2022]. |
| [2] | A. Vaisman and E. Zimányi, Data warehouse systems design and implementation (second edition), 2nd ed., 2022. |
| [3] | H. Akid and M. B. Ayedl, "Toward NoSQL Graph Data Warehouse for Big Social Data Analysis," 2016. |
| [4] | H. Akid, G. Frey, M. B. Ayed and N. Lachiche, "Performance of NoSQL Graph Implementations of Star vs. Snowflake Schemas," 2022. |
| [5] | C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen and D. Wilkins, "A Comparison of a Graph Database and a Relational Database," 2010. |
| [6] | N. Thu, V. Ngoc and L. Binh, "An approach to constructing a graph data repository for course recommendation based on IT career goals in the context of big data," in *2022 IEEE International Conference on Big Data (IEEE BigData 2022)*, Osaka, Japan, 2022. |
| [7] | E. Bernier, Y. Bédard, T. Badard and F. Hubert, "UMapIT© (Unrestricted Mapping Interactive Tool): Merging the datacube paradigm with an occurrence-based approach to support on-demand web mapping," 2007. |
| [8] | A. Ghrab, O. Romero, S. Jouili and S. Skhiri, "Graph BI & Analytics: Current State and Future Challenges," 2018. |
| [9] | B. P. Xuân and T. N. Lê, "Nghiên cứu các nền tảng phân tích dữ liệu đồ thị cho hệ thống tư vấn thời gian thực," 2022. |
| [10] | S. S. Owais and N. S. Hussein, "Extract Five Categories CPIVW from the 9V’s Characteristics of the Big Data," 2016. |
| [11] | R. Kiran, "What are the Characteristics of Big Data? | 5V's, Types, Benefits," 22 November 2022. [Online]. Available: https://www.edureka.co/blog/big-data-characteristics/. [Accessed 19 February 2023]. |
| [12] | S. S. Owais and N. S. Hussein, "Extract Five Categories CPIVW from the 9V’s Characteristics of the Big Data," 2016. |
| [13] | *Secret Sauce of Neo4j: Modeling and Querying Graphs.* [Film]. Neo4j, 2018. |
| [14] | "What is a Data Mart? (vs a Data Warehouse)," [Online]. Available: https://www.talend.com/resources/what-is-data-mart/. [Accessed 27 September 2022]. |
| [15] | "Phân biệt Data Warehouse, Data Mart và Data Lake - Tìm hiểu cùng FUNiX," 19 August 2021. [Online]. Available: https://funix.edu.vn/hoi-dap-cntt/phan-biet-data-warehouse-data-mart-va-data-lake/. [Accessed 11 November 2022]. |
| [16] | "Data Mart Architecture Defined - Best Practices and Key Concepts | Astera," [Online]. Available: https://www.astera.com/type/blog/data-mart-architecture/. [Accessed 25 9 2022]. |
| [17] | N. Bronson, Z. Amsden, G. Cabrera, P. Chakka, P. Dimov, H. Ding, J. Ferris, A. Giardullo, S. Kulkarni, H. Li, M. Marchukov, D. Petrov, L. Puzar, Y. Song, V. Venkataramani and Facebook, Inc., "TAO: Facebook’s Distributed Data Store for the Social Graph," 2013. |
| [18] | S. Sankar, "Did you mean "Galene"?," 5 June 2014. [Online]. Available: https://engineering.linkedin.com/search/did-you-mean-galene. [Accessed 3 October 2022]. |
| [19] | "FlockDB," [Online]. Available: https://en.wikipedia.org/wiki/FlockDB. [Accessed 3 October 2022]. |
| [20] | M. Zekri, M. Imen and A. Abdellatif, "A New Data Warehouse Approach Using Graph," 2011. |
| [21] | "Thực trạng nguồn nhân lực công nghệ thông tin ở Việt Nam," [Online]. Available: https://irdm.edu.vn/thuc-trang-nguon-nhan-luc-cong-nghe-thong-tin/. [Accessed 6 January 2023]. |
| [22] | G. Zhu, Y. Chen and S. Wang, "Graph-Community-Enabled Personalized Course-Job Recommendations with Cross-Domain Data Integration," 2022. |
| [23] | N. Li, N. Suri, Z. Gao, T. Xia, K. Börner and X. Liu, "Enter a Job, Get Course Recommendations," 2017. |
| [24] | A. Gugnani, V. Kasireddy and K. Ponnalagu, "Generating Unified Candidate Skill Graph for Career Path Recommendation," 2018. |
| [25] | "JobOntology," [Online]. Available: https://github.com/ptxhien/JobOntology?fbclid=IwAR1PmsBDiJeYL2t48udo-tqaGQTZAhSn-laNjXxzyBa9egxN43IYbljxTlY. [Accessed 20 February 2023]. |
| [26] | "graph DBMS," [Online]. Available: https://db-engines.com/en/ranking/graph+dbms. [Accessed 20 November 2022]. |
| [27] | "state of big data processing 2020," [Online]. Available: https://www.qubole.com/blog/state-of-big-data-processing-2020/https:/www.qubole.com/blog/state-of-big-data-processing-2020/. [Accessed 25 January 2023]. |
| [28] | "Apache Spark," [Online]. Available: https://spark.apache.org/. [Accessed 6 March 2023]. |
| [29] | "Apache Flink® — Stateful Computations over Data Streams | Apache Flink," [Online]. Available: https://flink.apache.org/. [Accessed 26 January 2023]. |
| [30] | "Apache Hadoop," [Online]. Available: https://hadoop.apache.org/. [Accessed 26 January 2023]. |
| [31] | "Apache Hive," [Online]. Available: https://hive.apache.org/. [Accessed 26 January 2023]. |
| [32] | "MongoDB," [Online]. Available: https://en.wikipedia.org/wiki/MongoDB. [Accessed 26 January 2023]. |
| [33] | "Documentation General Concepts," 21 December 2022. [Online]. Available: https://nlp.johnsnowlabs.com/docs/en/concepts. [Accessed 25 October 2022]. |
| [34] | A. Garg, "What is Spark DataFrame? - Spark DataFrame," 21 December 2022. [Online]. Available: https://intellipaat.com/blog/tutorial/spark-tutorial/spark-dataframe/. [Accessed 2 February 2023]. |
| [35] | "Neo4j Connector for Apache Spark v5.0.0," [Online]. Available: https://neo4j.com/docs/spark/current/. [Accessed 20 October 2022]. |