BỘ THÔNG TIN VÀ TRUYỀN THÔNG
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**Hoàng Minh Quang**

# NGHIÊN CỨU, PHÁT TRIỂN MỘT SỐ PHƯƠNG PHÁP KHAI PHÁ DỮ LIỆU TRÊN DỮ LIỆU CÓ CẤU TRÚC

Chuyên ngành : Hệ thống thông tin

Mã số: 09.48.01.04

TÓM TẮT LUẬN ÁN TIẾN SĨ KỸ THUẬT

Hà Nội - Năm 2020

Công trình được hoàn thành tại:

Học Viện Công Nghệ Bưu chính Viễn thông

Người hướng dẫn khoa học:

1. GS. TS. Vũ Đức Thi
2. GS. TSKH. Nguyễn Ngọc San

Phản biện 1: ……

Phản biện 2: ……

Phản biện 3: ……

Luận án được bảo vệ trước Hội đồng chấm luận án cấp Học viện họp tại:

Học viện Công nghệ Bưu chính Viễn thông

Vào hồi giờ …… ngày …… tháng …… năm ……

Có thể tìm hiểu luận án tại:

Thư viện Học viện Công nghệ Bưu chính Viễn thông

# DANH MỤC CÔNG TRÌNH CÔNG BỐ

[1] János Demetrovics, Hoang Minh Quang, Nguyen Viet Anh, and Vu Duc Thi. "An Optimization of Closed Frequent Subgraph Mining Algorithm". In: *Cybernetics and Information Technologies* 17.1 (2017), pp. 3–15.

[2] János Demetrovics, Hoang Minh Quang, Vu Duc Thi, and Nguyen Viet Anh. "An Efficient Method to Reduce the Size of Consistent Decision Tables". In: *Acta Cybernetica* 23.4 (2018), pp. 1039–1054. DOI: 10.14232/actacyb.23.4.2018.4.

[3] Hoang Minh Quang and Nguyen Ngoc Cuong. "Vấn đề phân loại đa nhãn cho đồ thị". In: *Proceeding of the eleventh National Symposium Fundamental and Applied Information Technology Research*. FAIR, Hanoi, Vietnam, 2018, pp. 567–574.

[4] Hoang Minh Quang, Vu Duc Thi, and Vu Thi Lan Anh. "Xây dựng cây quyết định từ bảng quyết định nhất quán". In: *Proceeding of the tenth National Symposium Fundamental and Applied Information Technology Research.* FAIR, Da Nang, Vietnam, 2017, pp. 633–640.

[5] Hoang Minh Quang, Vu Duc Thi, and Pham Quoc Hung. "Một số vấn đề về khai phá đồ thị con thường xuyên đóng". In: *Proceeding of the ninth National Symposium Fundamental and Applied Information Technology Research*. FAIR, Can Tho, Vietnam, 2016, pp. 471–479.

[6] Hoang Minh Quang, Vu Duc Thi, and Nguyen Ngoc San. "Some algorithms related to consistent decision table". In: *Journal of Computer Science and Cybernetics* 33.2 (2017), pp. 131–142.

[7] Hoang Minh Quang, Vu Duc Thi, Kieu Thu Thuy, Dao Van Tuyet, and Phan Trung Kien. "Khai phá cây con thường xuyên trên cơ sở dữ liệu weblogs". In: *Proceeding of the eighth National Symposium Fundamental and Applied Information Technology Research*. FAIR, Ha Noi, Vietnam, 2015, pp. 327–355.

# MỞ ĐẦU

## 1. TỔNG QUAN LUẬN ÁN VÀ LÝ DO CHỌN ĐỀ TÀI

Khai phá dữ liệu lớn là một xu hướng phát triển công nghệ mang tính cách mạng, ngày càng được ứng dụng rộng rãi, và đặc biệt còn nhiều tiềm năng phát triển trên toàn thế giới. Khai phá dữ liệu lớn có thể được ứng dụng để cải tiến công nghệ ở nhiều lĩnh vực quan trọng như: y tế, giao thông, tài chính, giáo dục, nhằm đem lại lợi ích trong việc hỗ trợ ra quyết định, cắt giảm chi phí, và tạo ra các sản phẩm, dịch vụ mới.

Mặc dù việc khai phá dữ liệu lớn đem lại giá trị to lớn và ý nghĩa, tuy nhiên, đây cũng là một lĩnh vực đòi hỏi công nghệ cao, đầu tư lớn, với nhiều thách thức. Nguyên nhân xuất phát từ hai đặc trưng cơ bản của dữ liệu lớn, đó là: tính lớn và tính đa dạng, phức tạp. Do độ lớn của dữ liệu, việc khai phá thường mất nhiều thời gian và chi phí, độ phức tạp tính toán của khai phá dữ liệu lớn thường là độ phức tạp hàm mũ. Hơn nữa, vì dữ liệu lớn và phức tạp, nên việc khai phá dữ liệu cần trích xuất được các thông tin cốt lõi để khai phá, thay vì xử lý cả tập hợp dữ liệu lớn, có nhiều dữ liệu dư thừa, không mang giá trị hữu ích. Do vậy, vấn đề cơ bản của xử lý dữ liệu lớn là cải tiến tốc độ xử lý dữ liệu và tăng giá trị của dữ liệu được khai phá.

Với ý nghĩa thực tiễn to lớn của ngành khai phá dữ liệu lớn, nhiều công trình khoa học đã tập trung nghiên cứu, phát triển các thuật toán nhằm cải tiến việc xử lý dữ liệu. Một số hướng nghiên cứu chính của các nhà khoa học trên thế giới trong việc khai phá dữ liệu như sau: đánh chỉ mục và truy vấn dữ liệu, tìm kiếm theo từ khóa, so khớp đồ thị, mô tả đồ thị lớn, khai phá các mẫu thường xuyên, phân cụm dữ liệu, phân lớp dữ liệu, khai phá các dữ liệu phát triển theo thời gian.

Trong luận án này, nghiên cứu sinh tập trung vào cả hai bài toán

cơ bản của ngành xử lý dữ liệu lớn là: tăng giá trị của dữ liệu và tăng tốc độ xử lý dữ liệu. Kết quả của luận án giúp nâng cao tính hiệu quả và giảm chi phí của việc khai phá dữ liệu lớn. Cụ thể, nghiên cứu sinh tập trung nghiên cứu, giải quyết hai bài toán: (i) một là các bài toán liên quan đến rút gọn thuộc tính, rút gọn đối tượng, giảm dữ liệu dư thừa, trích xuất được những dữ liệu nhỏ, đặc trưng, chính xác hơn, nhằm xác định giá trị cốt lõi trong tập hợp dữ liệu lớn và phức tạp, (ii) hai là bài toán tối ưu hóa tính toán, cải thiện tốc độ và chi phí tính toán trong khai phá dữ liệu có độ phức tạp tính toán lớn như độ phức tạp tính toán hàm mũ hay độ phức tạp tính toán trong thời gian không đa thức.

## 2. MỤC TIÊU - ĐỐI TƯỢNG - PHẠM VI NGHIÊN CỨU

### Mục tiêu nghiên cứu

Đặt mục tiêu giải quyết hai bài toán trên, nghiên cứu sinh nghiên cứu, phát triển một số phương pháp khai phá dữ liệu trên dữ liệu có cấu trúc, tập trung vào dữ liệu biểu diễn cấu trúc dạng bảng và dạng đồ thị. Đối với dữ liệu dạng bảng, mục tiêu nghiên cứu là các bài toán giảm dư thừa dữ liệu, rút gọn thuộc tính, rút gọn đối tượng để thu được tập dữ liệu nhỏ hơn trong khi vẫn bảo toàn được tính chất rút gọn thuộc tính, sinh cây quyết định trong khai phá dữ liệu lớn. Đối với biểu diễn dữ liệu dạng đồ thị, mục tiêu nghiên cứu là tối ưu tính toán các bài toán có độ phức tạp thời gian không đa thức xuống thời gian đa thức sử dụng một số ràng buộc dữ liệu để có thể khám phá tri thức từ dữ liệu trong thời gian chấp nhận được và các bài toán liên quan đến khai phá các tập dữ liệu mà dạng biểu diễn đồ thị còn gặp khó khăn trong khi đối với các dạng biểu diễn dữ liệu khác đã có phương pháp thực hiện.

### Đối tượng nghiên cứu

Trong luận án này, nghiên cứu sinh đặt trọng tâm khai phá dữ liệu

trên biểu diễn dữ liệu có cấu trúc dạng bảng quyết định nhất quán và biểu diễn đồ thị của cơ sở dữ liệu đồ thị như biểu diễn dữ liệu cấu trúc hóa học, biểu diễn dữ liệu sinh học, biểu diễn dữ liệu mạng máy tính, mạng xã hội. Trên tập dữ liệu được lựa chọn, nghiên cứu sinh phát triển một số thuật toán phục vụ khai phá dữ liệu lớn như giảm dư thừa, rút gọn dữ liệu hoặc tối ưu tính toán về độ phức tạp thời gian đa thức để đáp ứng thời gian khai phá dữ liệu cho phép đối với các thuật toán mà thông thường cần giải quyết trong độ phức tạp thời gian không đa thức.

**Phạm vi nghiên cứu**

Luận án tập trung vào hai đối tượng với phạm vi như: (i) bảng quyết định nhất quán với các bài toán tìm một rút gọn thuộc tính không heuristic, tìm một rút gọn đối tượng và sinh cây quyết định, và (ii) cơ sở dữ liệu giao tác đồ thị với bài toán khai phá đồ thị con thường xuyên đóng và phân loại đồ thị đa nhãn.

## 3. KẾT QUẢ - Ý NGHĨA KHOA HỌC VÀ THỰC TIỄN

Trong luận án, nghiên cứu sinh đã nghiên cứu cải tiến một số phương pháp khai phá dữ liệu đối với biểu diễn dữ liệu có cấu trúc dạng bảng và dạng đồ thị. Các kết quả đạt được bao gồm:

1. *Nghiên cứu rút gọn thuộc tính bảng quyết định nhất quán* Tìm được một rút gọn thuộc tính trong thời gian đa thức không sử dụng heuristic như các phương pháp tìm một rút gọn thuộc tính khác.

2. *Nghiên cứu rút gọn đối tượng bảng quyết định nhất quán* Tìm được một rút gọn đối tượng trong thời gian đa thức mà vẫn bảo toàn quá trình tìm tất cả các rút gọn thuộc tính.

3. *Nghiên cứu cây quyết định* Cải tiến phương pháp sinh cây quyết

định thực hiện nhanh hơn quá trình sinh cây quyết định của thuật toán ID3.

4. *Nghiên cứu khai phá đồ thị con thường xuyên đóng* Chứng minh vấn đề đẳng cấu đồ thị con giải quyết trong thời gian đa thức trong khai phá đồ thị con thường xuyên đóng trong khi các thuật toán khai phá đồ thị con thường xuyên đóng khác chưa giải quyết được vấn đề đẳng cấu đồ thị con trong thời gian đa thức.

5. *Nghiên cứu phân loại đa nhãn cho đồ thị* Xây dựng độ đo trên dàn giao khái niệm áp dụng cho phân loại đa nhãn đồ thị sử dụng lý thuyết Dempster-Shafer, trong khi các công trình phân loại đa nhãn theo lý thuyết Dempster-Shafer khác phải xây dựng độ đo dựa trên biểu diễn véctơ mà đồ thị không có biểu diễn véctơ.

Các kết quả nghiên cứu của nghiên cứu sinh đều có chứng minh tính đúng đắn và đầy đủ đã thể hiện ý nghĩa khoa học của luận án. Ngoài ra, các kết quả này có thể áp dụng cho cả các vấn đề nghiên cứu lẫn thực tiễn, các thuật toán nghiên cứu sinh đề xuất được áp dụng cho các bộ dữ liệu UCI dataset hoặc NCI dataset như Balance scale, Kr-vs-kp, Breast cancer, Car, Tic-tac-toe, Molecula, HIV AIDS, Chemical compound, ... trong một số kết quả thử nghiệm.

## 3. CẤU TRÚC LUẬN ÁN

Cấu trúc luận án có 3 chương như sau:

- Chương 1. Kiến thức chuẩn bị: Chương này trình bày một số các định nghĩa cơ sở, các định lý của các lý thuyết sẽ được áp dụng vào các phương pháp phát triển các thuật toán trong luận án này như lý thuyết tập thô, lý thuyết cơ sở dữ liệu quan hệ,

lý thuyết đồ thị, lý thuyết phân tích khái niệm chính thức, lý thuyết về độ tin cậy, lý thuyết Dempster-Shafer.

- Chương 2. Chương này trình bày chi tiết về một số phương pháp nghiên cứu sinh đề xuất trong việc phát triển các thuật toán khai phá dữ liệu trên biểu diễn dữ liệu có cấu trúc dạng bảng như rút gọn đối tượng trong thời gian đa thức, rút gọn thuộc tính không heuristic trong thời gian đa thức và sinh cây quyết định với thời gian thực hiện nhanh hơn thuật toán ID3, đồng thời nghiên cứu sinh cũng chứng minh tính đúng đắn và đầy đủ của các phương pháp này.

- Chương 3. Chương này trình bày một số phương pháp nghiên cứu sinh đề xuất về khai phá dữ liệu trên biểu diễn dữ liệu cấu trúc dạng đồ thị như bài toán khai phá đồ thị con thường xuyên đóng và phân loại đồ thị đa nhãn theo lý thuyết Dempster-Shafer. Trong bài toán khai phá đồ thị con thường xuyên đóng, nghiên cứu sinh đề xuất phương pháp xác định đẳng cấu đồ thị con trong thời gian đa thức và trong bài toán phân loại đa nhãn đồ thị, nghiên cứu sinh đề xuất độ đo khoảng cách trên dàn giao khái niệm phục vụ cho quá trình phân loại, đồng thời nghiên cứu sinh cũng chứng minh tính đúng đắn và đầy đủ của các phương pháp này.

# 1 KIẾN THỨC CHUẨN BỊ

## 1.1 Lý thuyết cơ sở dữ liệu quan hệ

Phần này trình bày một số định nghĩa trong cơ sở dữ liệu quan hệ. Kết hợp với các định nghĩa của lý thuyết tập thô, các định nghĩa về tập bằng nhau, hệ bằng nhau cực đại, khóa, phản khóa góp phần thực

hiện nhiệm vụ rút gọn thuộc tính và rút gọn đối tượng trên bảng quyết định nhất quán.

## 1.2 Lý thuyết tập thô

Phần này trình bày một số khái niệm cơ bản về lý thuyết tập thô như bảng thông tin, bảng quyết định, bảng quyết định nhất quán, quan hệ bất khả phân biệt, phân hoạch, lớp tương đương, rút gọn, ma trận phân biệt, tập lõi. Các định nghĩa này được áp dụng trong bài toán tìm một rút gọn thuộc tính trong thời gian đa thức, tìm rút gọn đối tượng trong thời gian đa thức và xây dựng cây quyết định từ bảng quyết định nhất quán thu gọn cả hai chiều ngang và dọc dựa trên rút gọn thuộc tính và rút gọn đối tượng.

## 1.3 Lý thuyết đồ thị

Phần này, nghiên cứu sinh trình bày một số định nghĩa về đồ thị phục vụ cho thuật toán khai phá đồ thị con thường xuyên đóng và giải quyết bài toán con của nó là đẳng cấu đồ thị con trong thời gian đa thức với ràng buộc về sử dụng máy truy cập ngẫu nhiên, tính có thứ tự của tập nhãn của đỉnh và cạnh.

## 1.4 Tập có thứ tự và dàn giao (lattices)

Tập có thứ tự và dàn giao là các khái niệm quan trọng trong việc xác định mối liên quan giữa hai phần tử trong một tập hợp các phần tử. Các khái niệm này là nền tảng xây dựng dàn giao khái niệm.

## 1.5 Phân tích khái niệm chính thức (FCA)

Phần này trình bày một số định nghĩa về ngữ cảnh chính thức, khái niệm chính thức, mối quan hệ cha - con giữa các khái niệm chính thức và dàn giao khái niệm. Từ những khái niệm này, nghiên cứu sinh đề xuất xây dựng độ đo tương tự giữa hai đồ thị trên dàn giao khái niệm phục vụ giải quyết bài toán phân loại đa nhãn trên đồ thị.

## 1.6 Biến đổi và đồng biến đổi Mobius

Biến đổi và đồng biến đổi Mobius được nghiên cứu sinh sử dụng trong việc xây dựng các hàm như hàm cấp phát khối, hàm niềm tin theo lý thuyết độ tin cậy Dempster-Shafer từ mối quan hệ trên dàn giao khái niệm của các đồ thị để phục vụ bài toán phân loại đa nhãn đồ thị sử dụng lý thuyết hàm niềm tin Dempster-Shafer.

## 1.7 Lý thuyết Dempster-Shafer

Phần này trình bày một số khái niệm cơ bản của lý thuyết Dempster-Shafer. Áp dụng luật Dempster trong việc kết hợp các hàm cấp phát khối và các hàm niềm tin thông qua tập các láng giềng của các đồ thị theo độ đo dàn giao khái niệm để xác định tập nhãn cho một đồ thị mới trong giải quyết bài toán phân loại đa nhãn cho đồ thị sử dụng lý thuyết Dempster-Shafer.

# 2 KHAI PHÁ DỮ LIỆU DẠNG BẢNG

Nội dung của chương này dựa trên các công trình số [0], [0], [0] trong danh mục công trình công bố của nghiên cứu sinh.

## 2.1 Rút gọn thuộc tính không heuristic

Tìm các rút gọn từ bảng quyết định là một trong các mục tiêu chính trong xử lý thông tin. Nhiều nghiên cứu tập trung vào rút gọn thuộc tính tức là làm giảm số cột trong bảng quyết định. Thật không may là tìm tất cả các rút gọn thuộc tính trong một bảng quyết định là vấn đề có độ phức tạp hàm mũ. Nghiên cứu sinh đề xuất một phương pháp đi tìm một rút gọn thuộc tính trong thời gian đa thức không theo phương pháp heuristic như các phương pháp khác. Thuật toán *AnAttributeReduct* nghiên cứu sinh đề xuất tìm một rút gọn thuộc tính được chứng minh tính đúng đắn và thực hiện thời gian đa thức.

**Algorithm 1:** AnAttributeReduct($DS$)

**Đầu vào:** $DS = (U, C \cup \{d\}, V, f)$

**Đầu ra :** $D \in RED(C)$

$E_r \leftarrow EqualitySet(DS)$;

$M_d \leftarrow MaximalEqualitySystem(DS, E_r)$;

$C = \{c_1, ..., c_n\}, H = C$;

**foreach** $i = 0; i < n; i++$ **do**

  **if** $\nexists B \in M_d : H - c_{i+1} \subseteq B$ **then**

    $H = H - c_{i+1}$;

  **end**

**end**

trả về $D = H(n)$ ($H(n)$ là $H$ khi vòng lặp kết thúc với $i = n = |C|$);

**Định lý 2.1.** $H(n) \in RED(C)$.

Độ phức tạp tính toán thời gian của thuật toán *AnAttributeReduct*($DS$) không lớn hơn $O(|C| \times |U|^4)$. Có thể thấy được rằng nếu thay đổi thứ

tự các phần tử của tập $C$ ở bước 3, có thể nhận được một rút gọn thuộc tính khác từ bảng quyết định nhất quán $DS$.

## 2.2 Rút gọn đối tượng bảng quyết định nhất quán

Dựa trên lý thuyết tập thô và lý thuyết cơ sở dữ liệu quan hệ nghiên cứu sinh đã đề xuất một phương pháp rút gọn các đối tượng của bảng quyết định nhất quán mà vẫn bảo toàn vấn đề tìm tập tất cả các tập rút gọn thuộc tính của bảng quyết định nhất quán.

**Bổ đề 2.2.** *Cho bảng quyết định nhất quán* $DS = (U, C \cup \{d\}, V, f)$ *với* $C = \{c_1, c_2, ..., c_n\}, U = \{u_1, u_2, ..., u_m\}$. *Xem* $DS$ *như một quan hệ* $r = \{u_1, u_2, ..., u_m\}$ *trên tập thuộc tính* $R = C \cup \{d\}$.
*Đặt* $E_r = \{E_{ij} : 1 \leqslant i < j \leqslant m\}$ *với* $E_{ij} = \{a \in R : a(u_i) = a(u_j)\}$.
*Đặt* $M_d = \{A \in E_r : d \notin A, \nexists B \in E_r : d \notin B, A \subset B\}$.
*Thì* $M_d = (K_d^r)^{-1}$ *với* $K_d^r$ *la họ các thuộc tính tối tiểu của thuộc tính* $\{d\}$ *trên quan hệ* $r$.

**Định nghĩa 2.1.** Một rút gọn đối tượng của bảng quyết định nhất quán $DS = (U, C \cup \{d\}, V, f)$ là một bảng quyết định nhất quán $DS' = (U', C \cup \{d\}, V, f)$, với $RED(C) = RED_U(C)$ và:
1) $U' \subseteq U$,
2) $RED_U(C) = RED_{U'}(C)$,
3) $RED_U(C) \neq RED_{U'-\{u\}}(C), \forall u \in U'$.

**Định lý 2.3.** $DS' = (U' = T(m), C \cup \{d\}, V, f)$ *trong thuật toán AnObjectReduct thỏa mãn ba điều kiện 1), 2) và 3) theo định nghĩa 2.1.*

Rõ ràng số bước tính toán $E_r$ theo định nghĩa hệ bằng nhau là ít hơn $|U|^2$. Số bước tính toán $M_d$ là ít hơn $|E_r|^2$ và $|E_r| \leqslant$

**Algorithm 2:** AnObjectReduct($DS$)

**Đầu vào:** $DS = (U, C \cup \{d\}, V, f)$

**Đầu ra :** $DS' = (U', C \cup \{d\}, V, f)$

```
E_r ← EqualitySet(DS);
M_d^U ← MaximalEqualitySystem(DS, E_r);
T = U = {u_1, ..., u_m};
foreach i = 0; i < |U|; i++ do
    if M_d^{T-u_{i+1}} = M_d^U then
        T = T - u_{i+1};
    end
end
trả về DS' = (U' = T(m), C ∪ {d}, V, f) (T(m) là T sau
    khi vòng lặp kết thúc với i = m = |U|);
```

$\frac{|U|(|U|-1)}{2}$. Do vậy, độ phức tạp thời gian tồi nhất của thuật toán *AnObjectReduct($DS$)* không lớn hơn $O(|U|^5)$. Có thể dễ dàng thấy rằng nếu thay đổi trật tự các phần tử của tập vũ trụ $U$, có thể tìm được một rút gọn đối tượng khác.

## 2.3 Xây dựng cây quyết định từ bảng rút gọn

Vấn đề xây dựng tất cả các cây quyết định từ bảng quyết định từ một bảng quyết định $DS$ là một vấn đề NP-đầy đủ bởi sẽ có $|C|!$ sự sắp xếp các thuộc tính để tạo cây quyết định. Các công trình xây dựng cây quyết định đều là heuristic dựa trên một số độ đo chẳng hạn như ID3 với Entropy và Gain. Nghiên cứu sinh đề xuất thuật toán sinh cây quyết định theo độ đo hàm chứa của quan hệ bất khả phân biệt.

**Định lý 2.4.** *Thủ tục $RecursiveNode(DS)$ là đúng đắn.*

**Procedure** RecursiveNode(DS)

1 $Node \leftarrow \varnothing$;
2 **if** $((|U| == 1)||(|C \cup \{d\}| == 0))$ **then**
3 $\quad Node \leftarrow U(d)$ (nút lá);
4 **else**
5 $\quad bestAttribute \leftarrow$
$\quad\quad max\,(\forall(e \in C \cup \{d\}) \sum (IND(e) \subseteq IND(d)))$;
6 $\quad remainAttributes \leftarrow (C \cup \{d\} - bestAttribute)$;
7 $\quad Node \leftarrow bestAttribute$;
8 $\quad Node.childs \leftarrow \{RecursiveNode(DS')\}$;
9 $\quad (DS' = (U' = U : Value(bestAttribute =$
$\quad\quad v), (C \cup \{d\}) - bestAttribute, V, f)),$
$\quad\quad (\forall v \in Value(bestAttribute))$;
10 **end**
11 trả về $Node$;

**Định lý 2.5.** *Thuật toán $IRDT(DS)$ là đúng đắn.*

Thực nghiệm kết quả đánh giá các thuật toán rút gọn thuộc tính (bảng 1) và rút gọn đối tượng (bảng 2) nghiên cứu sinh đề xuất trong chương khai phá dữ liệu dạng bảng. Các kết quả thực nghiệm được thực hiện nhanh với ngôn ngữ lập trình Nodejs, Javascript với một số tập dữ liệu dạng txt từ kho dữ liệu UCI.

Thực nghiệm chỉ ra rằng tốc độ tính toán của thuật toán $IRDT$ là nhanh vượt trội so với thuật toán ID3 (bảng 3). Có thể dễ dàng nhận thấy rằng vấn đề đếm số lượng phần tử trong các tập quan hệ bất khả phân biệt là các phép tính toán số nguyên nên rõ ràng nhanh hơn tính Entropy và tính Information Gain vốn là các công thức tính toán số thực.

**Algorithm 3:** IRDT($DS$)

**Đầu vào:** $DS = (U, C \cup \{d\}, V, f)$

**Đầu ra :** $DecisionTree(DS)$

1 $Root \leftarrow RecursiveNode(DS = (U, C \cup \{d\}, V, f))$;

2 trả về $Root$;

Bảng 1: Bảng thực hiện một rút gọn thuộc tính

| Tập dữ liệu | Thuộc tính gốc | Thuộc tính rút gọn | Thời gian(s) |
|---|---|---|---|
| Examples | 4 | 3 | 0.006 |
| Breast cancer | 9 | 8 | 0.161 |
| Balance | 4 | 3 | 0.248 |
| Car Evaluation | 6 | 5 | 0.673 |

# 3 KHAI PHÁ DỮ LIỆU ĐỒ THỊ

Nội dung chương này dựa trên các công trình số [0], [0], [0], [0] trong danh mục công trình công bố của nghiên cứu sinh.

## 3.1 Khai phá đồ thị con thường xuyên đóng

Nghiên cứu sinh đề xuất một phương pháp khai phá mẫu đồ thị con thường xuyên với việc kiểm tra đẳng cấu đồ thị con trong thời gian đa thức với ràng buộc gán nhãn và thứ tự nhãn của cả đỉnh và cạnh trong tất cả đồ thị của cở sở dữ liệu đồ thị. Thuật toán mới do nghiên cứu sinh đề xuất cho việc khai phá các đồ thị con thường xuyên đóng dựa trên chiến lược nhãn chuẩn hóa, mô hình máy truy cập ngẫu nhiên (RAM) hoặc mô hình von Neumann và cách tiếp cận

Bảng 2: Bảng thực hiện rút gọn đối tượng

| Tập dữ liệu | Đối tượng gốc | Đối tượng rút gọn | Thời gian(s) |
|---|---|---|---|
| Examples | 14 | 6 | 0.005 |
| Breast cancer | 286 | 2 | 0.158 |
| Balance | 625 | 6 | 0.171 |
| Car Evaluation | 1728 | 9 | 0.771 |

Bảng 3: Bảng so sánh tốc độ thực hiện IDRT và ID3 (millisecond)

| Datasets (Atts/Objs) | ID3 (ms) | $IRDT$ (ms) |
|---|---|---|
| Examples (4/14) | 3 | 1 |
| Breast cancer (9/286) | 53 | 13 |
| Car Evaluation (6/1728) | 64 | 30 |

Apriori với tính đóng nhằm giảm bớt số lượng ứng viên và các đồ thị con thường xuyên được sinh ra. Trong thuật toán mới, bài toán đồ thị con đẳng cấu được giải quyết trong thời gian đa thức so với giải quyết trong thời gian không đa thức trong các thuật toán hiện có. Thêm vào đó nghiên cứu sinh cũng chỉ ra tính đúng đắn và độ phức tạp của thuật toán mới được đề xuất.

**Nhãn chuẩn hóa**

Trong các công trình nghiên cứu của Huan, Yan 2003 đã chỉ ra việc sử dụng biểu diễn duy nhất cho một đồ thị làm giảm thời gian thực hiện khai phá đồ thị con thường xuyên.

**Sinh tập ứng viên**

Trong thuật toán mới, PSI-CFSM, xác định tất cả các $FS_2^i$, với

mọi đồ thị con đóng thường xuyên từ tập $CS^i_{k-1}$, xây dựng tập đồ thị con ứng viên $C^i_k$ với độ phức tạp thời gian đa thức.

**Kiểm tra đồ thị con đẳng cấu**

Thuật toán PSI-CFSM cải tiến các bước kiểm tra đẳng cấu đồ thị con bằng cách sử dụng kiểm tra đẳng cấu đồ thị con theo tìm kiếm nhị phân trong mô hình máy truy cập ngẫu nhiên. Trong lý thuyết độ phức tạp tính toán, sự phức tạp về thời gian của tìm kiếm nhị phân là $O(logn)$ trong đó $n$ là số lượng ứng viên đồ thị con. Giả sử lực lượng của các đồ thị con ứng viên là $2^n$ thì số bước của phép kiểm tra đẳng cấu đồ thị con bằng cách tìm kiếm nhị phân trên mô hình máy truy cập ngẫu nhiên là $log_2 2^n = n$ và độ phức tạp của thời gian là $O(n)$.

---

**Procedure** TestIsomorphism($g \in C^j_k, C^i_k$)

---

**Đầu vào:** $g \in C^j_k, C^i_k$
**Đầu ra :** $true \vee false$
1 $b \leftarrow$ BinarySearch($\{code(CAM(g' \in C^i_k))\}, code(CAM(g)), 0, |C^i_k|$);
2 **if** $b > 0$ **then**
3 | trả về true;
4 **else**
5 | trả về false;
6 **end**

---

**Bổ đề 3.1.** *Độ phức tạp tính toán của TestIsomorphism là* $O(log_2|C^i_k|)$

**Bổ đề 3.2.** *Thủ tục* $TestIsomorphism(g \in C^j_k, C^i_k)$ *là đúng đắn.*

**Thuật toán PSI-CFSM**

Trong thuật toán PSI-CFSM, bước đầu tiên là xây dựng mảng được sắp xếp thứ tự theo trật tự của mã CAM của các đồ thị con

với 2 đỉnh (chỉ có một cạnh) 2-subgraph của đồ thị $G_i$ trong cơ sở dữ liệu đồ thị $\mathbb{GD}$. Mảng được sắp xếp thứ tự này ký hiệu là $C_2^i$, $C_2 = \{C_2^i\}$. Với mỗi phần tử $u$ trong $C_2^i$, so sánh $codeCAM(u)$ với $codeCAM(v), v \in \{C_2^j = C_2 - C_2^i\}$. Nếu $code(CAM(u)) = code(CAM(v))$ thì tăng độ hỗ trợ của $u$ lên 1. Nếu $sup_u \geqslant \sigma$ thì đặt $u$ vào trong $FS_2$, $FS_2^i$. $FS_2$ ($FS_2^D$) là tập các đồ thị con thường xuyên 2-subgraphs của cơ sở dữ liệu đồ thị $\mathbb{GD}$ và $FS_2^i$ là tập các đồ thị con thường xuyên 2-subgraphs của đồ thị $G_i \in \mathbb{GD}$. Xây dựng một vòng lặp với $k \geqslant 3$ để tính $C_k^i, FS_k, FS_k^i, CS_k, CS_k^i$ dựa trên thuật toán PSI-CFSM.

**Định lý 3.3.** *Thuật toán PSI-CFSM là đúng đắn.*

## 3.2 Phân loại đa nhãn cho đồ thị

Denoeux 2012 đã đề xuất một phương pháp để giảm độ phức tạp tính toán trong thao tác và kết hợp các hàm khối, khi các hàm niềm tin được xác định trên một tập con phù hợp của *khung phân biệt* được kết hợp với cấu trúc dàn giao.

**Xây dựng dàn giao khái niệm**

Xây dựng dàn giao cho các đồ thị $g_i \in \mathbb{GD}$ sử dụng một *bảng ngữ cảnh chính thức* theo định nghĩa ngữ cảnh chính thức bằng cách xây dựng tập tất cả các đồ thị con thường xuyên đóng $CS$ của cơ sở dữ liệu đồ thị $\mathbb{GD}$ và coi tập $CS$ là tập các thuộc tính còn cơ sở dữ liệu đồ thị tập đối tượng. Mối quan hệ giữa tập đối tượng và tập thuộc tính thể hiện qua việc một đồ thị $G_i \in \mathbb{GD}$ có chứa một đồ thị con thường xuyên đóng $g_j \in CS$ thì đồ thị $G_i$ và đồ thị con thường xuyên đóng $g_j$ là có mối quan hệ với nhau.

Từ bảng ngữ cảnh chính thức, tìm ra tất cả các *khái niệm chính thức*, xây dựng được một dàn giao khái niệm IcebergLattice.

**Algorithm 4:** PSI-CFSM($\mathbb{GD}$, $\sigma$ = min_sup)

**Đầu vào:** Cơ sở dữ liệu đồ thị $\mathbb{GD}$, $\sigma$ = min_sup

**Đầu ra :** $CS_2, CS_3, ..., CS_k$, các tập đồ thị con thường xuyên đóng theo mức

```
1  Xây dựng mảng có thứ tự theo code(CAM) của C_2^i;
2  foreach u ∈ C_2^i do
3      TestIsomorphism(u, C_2^j) và tìm sup_u ⩾ σ để đặt u vào
       trong FS_2^i, FS_2^D, CS_2^i và CS_2;
4  end
5  k ← 3;
6  while Combine(∀CS_{k-1}^i, ∀FS_2^i) is not null do
7      Xây dựng mảng có thứ tự theo code(CAM) của C_k^i;
8      foreach u ∈ C_k^i do
9          TestIsomorphism(u, C_k^j) và tìm sup_u ⩾ σ để đặt u
           vào trong CS_k^i và CS_k;
10         Kiểm tra v ∈ CS_{k-1}^i nếu sup_v = sup_u thì xóa v khỏi
           CS_{k-1};
11     end
12     k ← k + 1;
13 end
```

Dựa trên định nghĩa dàn giao, dàn giao khái niệm thì Iceberglattice $CL$ sẽ luôn có phần tử *cận trên nhỏ nhất* và *cận dưới lớn nhất* cho mỗi cặp phần tử trên dàn giao khái niệm. Từ dàn giao khái niệm, định nghĩa một độ đo dựa trên khoảng cách tính theo số lượng cạnh tính từ phần từ nhỏ nhất cận dưới $lub(x, y)$ đến mỗi đỉnh $x, y \in CL$ trên dàn giao khái niệm gọi là $d(x, y)$.

**Định nghĩa 3.1.** Đường đi giữa hai đỉnh $x, y$ trên dàn giao khái niệm $CL$ là tổng các đường đi ngắn nhất từ $lub(x, y)$ đến $x$ và từ $lub(x, y)$

đến $y$.

**Bổ đề 3.4.** *Đường đi giữa hai đỉnh $x, y$ theo định nghĩa 3.1 là đường đi ngắn nhất.*

**Định nghĩa 3.2.** Cho $CL$ là một dàn giao khái niệm, độ đo tương tự giữa hai đồ thị $g_i, g_j \in \mathbb{GD}$ là đường đi giữa hai đỉnh khái niệm chính thức chứa $g_i, g_j$.

$$d(g_i, g_j) = |shortest\_path(c(g_i), c(g_j))|$$

với $c(g_i), c(g_j)$ là các khái niệm chính thức của các đồ thị $g_i, g_j$ trong ngữ cảnh chính thức của cơ sở dữ liệu đồ thị $\mathbb{GD}$.

**Định lý 3.5.** *$d(g_i, g_j)$ thỏa mãn tính chất của độ đo tương tự theo khoảng cách.*

**Thuật toán phân loại đa nhãn đồ thị**

Thuật toán phân loại đa nhãn cho đồ thị được xây dựng theo phương pháp k-láng giềng gần nhất để xác định tập nhãn cho đồ thị $g_n \in \mathbb{GD}$ chưa có nhãn với mọi đồ thị $G_i \in \mathbb{GD}$ đã được gán nhãn $L_i \subseteq L$. Tương ứng với mỗi đồ thị $g_i \in kNN(g_n)$ sẽ là một hàm niềm tin với khoảng nhãn $[A_i, B_i]$ được xác định theo dàn giao khái niệm $CL$ với $A_i$ là tập nhãn của $g_i$ và $B_i$ là tập nhãn của $lub(g_i, g_n)$. Độ đo tương tự $d$ và $x_i$ là một phần tử của tập k láng giềng gần nhất có tập nhãn nằm trong khoảng $[A_i, B_i]$ (poset hữu hạn cục bộ) thì một mục bằng chứng có thể được mô tả như hàm khối sau:

$$m_i([A_i, B_i]) = \alpha_i, \tag{1}$$

$$m_i([\varnothing_\Gamma, \Gamma]) = 1 - \alpha_i \tag{2}$$

với $\alpha_i$ là độ đo tương tự dựa trên công thức (3.2) theo tỉ lệ đối với tổng khoảng cách tất cả các đồ thị $g_k$ tới $g_n$.

**Algorithm 5:** DSMLGC($DS$)

**Đầu vào:** $\mathbb{GD}$, $L$, $g_x$

**Đầu ra :** $A \subseteq L$ là tập nhãn của $g_x$

1 Xây dựng dàn giao khái niệm $IcebergLattice$ cho $\mathbb{GD}$ và $g_x$;

2 Xác định k-láng giềng của $g_x$ trên $IcebergLattice$ là tập $kNN(g_x)$;

3 Áp dụng luật Dempster-Shafer tìm tập nhãn cho $g_x$ từ $kNN(g_x)$;

Denoeux đề xuất luật để xác định tập nhãn cho đối tượng $x$. Cho $\hat{Y}$ là tập nhãn dự đoán sẽ được gán cho $x$. Để quyết định mỗi nhãn $\theta \in \Gamma$ được gán cho $x$ hay không, hai số lượng được tính là cấp độ hàm niềm tin $bel([\{\theta\}, \Gamma])$, $\hat{Y}$ là tập nhãn đúng chứa $\theta$, và cấp độ hàm niềm tin $bel([\varnothing, \{\bar{\theta}\}])$ mà không chứa $\theta$. Tập nhãn dự đoán được gán $\hat{Y}$ được xác định như sau:

$$\hat{Y} = \{\theta \in \Gamma \mid bel([\{\theta\}, \Gamma]) \geqslant bel([\varnothing, \{\bar{\theta}\}])\}. \tag{3}$$

Thực nghiệm chứng tỏ rằng phương pháp khai phá đồ thị con thường xuyên đóng PSI-CFSM của nghiên cứu sinh đề xuất tối ưu về mặt thời gian tính toán hơn gSpan nhờ vấn đề xác định đẳng cấu đồ thị con trong thời gian đa thức (bảng 4). Sử dụng các bộ dữ liệu Chemical Compound đi kèm với thuật toán gSpan và bộ dữ liệu nghiên cứu sinh tự sinh trong phần ví dụ, cùng với việc đặt các ngưỡng độ hỗ trợ tối thiểu khác nhau để so sánh thời gian thực hiện 2 thuật toán PSI-CFSM và gSpan. Kết quả được cho trong bảng sau:

Bảng 4: Khai phá đồ thị con thường xuyên (đơn vị thời gian: giây)

| Ngưỡng (xuất hiện) | Thuật toán | Dữ liệu 4 | Dữ liệu 50 |
|---|---|---|---|
| 2 | gSpan | 0.07s | 1120s |
| 2 | PSI-CFSM | 0.027s | 66.2s |
| 5 | gSpan | 0.0s | 3.26s |
| 5 | PSI-CFSM | 0.006s | 2.986s |
| 10 | gSpan | 0.0s | 1.74s |
| 10 | PSI-CFSM | 0.006s | 1.42s |

# KẾT LUẬN

Dữ liệu lớn dẫn đến nhu cầu rút gọn dữ liệu để giảm không gian lưu trữ và tối ưu thời gian tính toán. Các công trình nghiên cứu tập trung vào tìm các rút gọn thuộc tính theo lý thuyết tập thô của Pawlak trên bảng quyết định. Tìm tất cả các rút gọn thuộc tính có độ phức tạp thời gian hàm mũ $O(m * 2^n)$ với $m$ là số lượng đối tượng và $n$ là số lượng thuộc tính của bảng quyết định nhất quán. Luận án phát hiện phương pháp tìm một rút gọn đối tượng $m' < m$ trong thời gian đa thức mà vấn đề tìm tất cả các rút gọn đối tượng được bảo toàn. Theo đó, độ phức tạp tính toán tìm tất cả các rút gọn thuộc tính chỉ còn là $O(m' * 2^n)$ và giảm không gian lưu trữ dữ liệu đặc biệt đối với dữ liệu lớn. Ngoài ra, để giảm độ phức tạp tính toán hàm mũ trong vấn đề sinh luật quyết định, sinh cây quyết định thì các nghiên cứu công bố tìm một rút gọn thuộc tính heuristic trong thời gian đa thức. Thêm vào đó luận án thành công tìm một rút gọn thuộc tính trong thời gian đa thức không heuristic và một phương pháp cải tiến sinh cây quyết định có tốc độ thực hiện nhanh hơn thuật toán sinh cây quyết định ID3 trên bảng quyết định nhất quán. Trong luận án, các đề xuất của

nghiên cứu sinh được chứng minh đúng đắn và đầy đủ cùng với thực nghiệm chứng tỏ thuật toán sinh cây quyết định của nghiên cứu sinh nhanh hơn thuật toán ID3.

Dữ liệu lớn là dữ liệu được thu thập từ nhiều miền, nhiều lĩnh vực do đó có đa dạng cấu trúc biểu diễn khác nhau. Các thuật toán khai phá dữ liệu chỉ có thể khai phá dữ liệu trên một tập dữ liệu thống nhất về kiểu dạng biểu diễn. Các cấu trúc dữ liệu khác nhau có thể biểu diễn dữ liệu dưới dạng đồ thị để thống nhất kiểu dạng cho các mục đích khai phá dữ liệu. Tuy nhiên, khai phá dữ liệu đồ thị có độ phức tạp thời gian không đa thức thậm chí là độ phức tạp hàm mũ. Trong luận án này, nghiên cứu sinh tập trung vào khai phá dữ liệu đồ thị con thường xuyên và phân loại đa nhãn đồ thị. Đối với bài toán khai phá đồ thị con thường xuyên, một vấn đề nổi cộm là xác định đẳng cấu đồ thị con thông thường có độ phức tạp không đa thức đầy đủ. Luận án đã giải quyết khai phá đồ thị con thường xuyên đóng bằng thuật toán PSI-CFSM trong đó vấn đề xác định đẳng cấu đồ thị con trong thời gian đa thức bằng cách áp dụng một số điều kiện ràng buộc về nhãn chuẩn hóa, máy truy cập ngẫu nhiên. Đối với bài toán phân loại đa nhãn, các mô hình phân loại đa nhãn áp dụng lý thuyết Dempster Shafer tăng độ chính xác phân loại và giảm thời gian tính toán không áp dụng được cho biểu diễn dữ liệu đồ thị do đồ thị thiếu biểu diễn dạng véctơ. Luận án thực hiện xây dựng dàn giao khái niệm dựa trên tập đồ thị con thường xuyên đóng của tập dữ liệu đồ thị để từ đó xác định độ đo khoảng cách giữa các đồ thị và dựa vào độ đo khoảng cách này để phân loại đa nhãn cho đồ thị theo lý thuyết Dempster Shafer. Trong luận án, các đề xuất của nghiên cứu sinh về xác định đẳng cấu đồ thị con và xác định độ đo khoảng cách trên dàn giao khái niệm được chứng minh tính đúng đắn và đầy đủ cùng với thực nghiệm chứng tỏ thuật toán PSI-CFSM tối ưu thời gian hơn so với thuật toán gSpan trong khai phá đồ thị con thường xuyên.