BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

TRẦN PHI LỰC

NGHIÊN CỨU MỘT SỐ THUẬT TOÁN GIA TĂNG LỰA CHỌN THUỘC TÍNH TRÊN BẢNG QUYẾT ĐỊNH ĐỘNG THEO TIẾP CẬN TẬP MỜ SỬ DỤNG LÁT CẮT α

Ngành Hệ thống thông tin
Mã số: 8480104

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ HỆ THỐNG THÔNG TIN

NGƯỜI HƯỚNG DẪN:
1. TS. Đặng Trọng Hợp

Hà Nội – 2024

LỜI CAM ĐOAN

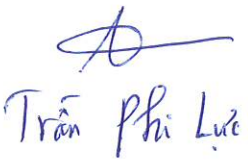
Tôi là Trần Phi Lực, học viên cao học lớp Cao học hệ thống thông tin khóa 12. Tôi cam đoan rằng đề án thạc sĩ mang tựa đề "Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α" được trình bày dưới đây là công trình nghiên cứu của chính tôi dưới sự hướng dẫn của TS. Đặng Trọng Hợp.

Các nội dung nghiên cứu và kết quả trong đề tài này là trung thực và chưa từng được ai công bố trong bất cứ công trình nghiên cứu nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi trong phần tài liệu tham khảo. Tôi cam đoan rằng không có bất kỳ vi phạm nào đối với các quy định đạo đức nghiên cứu khoa học trong quá trình thực hiện luận án. Các tài liệu tham khảo được trích dẫn đúng nguồn gốc và được sử dụng một cách hợp lý.

Tôi hiểu rõ rằng nếu phát hiện bất kỳ sai sót, vi phạm hoặc gian lận nào trong đề án của mình, tôi sẽ chịu trách nhiệm trước pháp luật và có thể bị xem xét lại về bằng cấp đã đạt được. Tôi viết cam đoan này và tôi hoàn toàn chịu trách nhiệm về tính chính xác và trung thực của công trình nghiên cứu này.

Hà Nội, ngày.....tháng.....năm 2024

Tác giả



Trần Phi Lực

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
RGTT	Attribute reduction	Rút gọn thuộc tính
BQĐ	Decision table	Bảng quyết định
TĐT	Object set	Tập đối tượng
TTM	The rough fuzzy set	Tập thô mờ
TRG	The reduced set	Tập rút gọn
DS	**D**ecision **s**ystem\ Decision table	Hệ thống quyết định
IS	**I**nformation **s**ystem	Hệ thống thông tin

DANH MỤC HÌNH VẼ

DANH MỤC CÁC BẢNG BIỂU

MỞ ĐẦU

I. Sự cần thiết triển khai đề tài

Lựa chọn thuộc tính là một bước trong quá trình tiền xử lý dữ liệu nhằm loại bỏ các thuộc tính dư thừa, không cần thiết để tăng tính dễ hiểu cho luật và hiệu quả cho các mô hình phân lớp. Trên thế giới, các nghiên cứu về lựa chọn thuộc tính hiện nay đang trở nên rất sôi động. Một trong những cách tiếp cận có thể nói tới là các phương pháp rút gọn thuộc theo hướng tiếp cận của lý thuyết tập thô [1]. Tuy nhiên, các phương pháp RGTT theo hướng tiếp cận này chỉ thực hiện được trên các BQĐ có miền giá trị rời rạc. Đối với các BQĐ có miền giá trị số, các phương pháp này phải chia thành nhiều khoảng tương ứng với các giá trị phân loại. Việc không thực hiện bước rời rạc hóa dữ liệu có thể dẫn đến mất mát thông tin quan trọng trên các BQĐ và gây ra sự suy giảm về hiệu quả của các mô hình phân loại. Để giải quyết vấn đề này, Dübois và đồng nghiệp [2] đã đề xuất một mô hình gọn trực tiếp trên BQĐ gốc với miền giá trị số, mà không cần thực hiện bước rời rạc hóa dữ liệu. Mô hình này được gọi là mô hình TTM (fuzzy rough set). Theo các phân tích về TTM, các nhà nghiên cứu đã xây dựng nhiều phương pháp RGTT trực tiếp trên BQĐ gốc có miền giá trị số sử dụng nhiều độ đo khác nhau. Với BQĐ cố định, các phương pháp điển hình là sử dụng hàm thuộc mờ [3, 4], miền dương mờ [5, 6], entropy thông tin mờ [7, 8], khoảng cách mờ [9, 10] và một số phương pháp khác [11, 12, 13]. Kết quả thực nghiệm trong các công bố nêu trên cho thấy, các thuật toán tìm TRG theo tiếp cận TTM nâng cao độ chính xác phân lớp so với các thuật toán theo tiếp cận tập thô truyền thống. Tuy nhiên, Hung và các cộng sự trong [14] trình bày, các phương pháp RGTT theo tiếp cận TTM không hiệu quả khi xử lý trên các BQĐ nhiễu và không nhất quán. Ngoài ra, trong xu thế bùng nổ của dữ liệu, các BQĐ có số tính chất vô cùng lớn. Hơn nữa, các BQĐ thay đổi liên tục, bổ sung với các trường hợp như tăng thêm hay bớt đi TĐT. Ví dụ điển hình như bài toán chẩn đoán bệnh trong lĩnh vực y tế, chẩn đoán các triệu chứng lâm

sàng dựa trên rất nhiều các chỉ số xét nghiệm. Số lượng bệnh nhân liên tục gia tăng theo thời gian dẫn tới quá trình xây dựng các mô hình phân lớp nhằm hỗ trợ bác sĩ trong việc chẩn đoán gặp rất nhiều khó khăn. Do vậy, để đưa ra một mô hình phân lớp có lợi, vấn đề đặt ra là phải giải quyết bài toán RGTT trên các BQĐ lớn và có sự di động về đối tượng.

Từ những khó khăn và thách thức đã nêu, đề tài **"Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α"** được lựa chọn như một hướng đi mới và đầy tiềm năng trong việc phát triển các thuật toán lựa chọn thuộc tính.

II. Mục tiêu nghiên cứu của đề tài

- Đề tài tìm hiểu, đề xuất các thuật toán gia tăng tìm TRG của BQĐ động dựa trên TTM theo tiếp cận tập mờ sử dụng lát cắt α nhằm giảm bớt thuộc tính TRG và tăng độ chính xác, giảm độ phức tạp của mô hình khai phá dữ liệu.

- Đề tài cung cấp một chương trình tính toán xác định tập thuộc tính rút gọn trên các bộ dữ liệu có sự biến động về số lượng các bản ghi (TĐT).

- Đề tài cũng trình bày một số phân tích để chứng minh tính hiệu quả của thuật toán trên các bộ dữ liệu khác nhau thông qua các tiêu chuẩn đánh giá về độ chính xác phân lớp và thời gian tính toán.

- Thực hiện so sánh, đánh giá về độ chính xác và tốc độ thực hiện của thuật toán so với các thuật toán nghiên cứu tước đó.

III. Phạm vi và nội dung nghiên cứu

Phạm vi của nghiên cứu này sẽ chỉ tập trung vào các phương pháp rút gọn dựa trên lý thuyết tập thô và các mở rộng, đặc biệt là tập mờ sử dụng lát cắt α với những hiệu quả mà nó mang lại trong bài toán RGTT. Có thể nói, tập mờ là một trong những công cụ rất mạnh và được ứng dụng vào rất nhiều bài toán về khai phá dữ liệu trong những năm trở lại gần đây. Tuy nhiên, cách tiếp cận này còn mới và chưa thực sự được quan tâm. Nghiên cứu này hy vọng sẽ là một bước tiến trong việc cải thiện các phương pháp RGTT theo hướng tiếp

cận tập thô và các mô hình mở rộng khi mang đến một công cụ hữu hiệu trong việc tìm kiếm các tập con thuộc tính trên các BQĐ, đặc biệt là các BQĐ có tính nhiễu, không nhất quán và có sự bổ sung cũng như loại bỏ TĐT theo thời gian. Đề tài này được nhóm nghiên cứu trình bày dựa trên cơ sở của nhiều nghiên cứu trước đây, kết hợp với các thực nghiệm để đánh giá và so sánh trên nhiều thuật toán nhằm chứng minh tính hiệu quả từ các phương pháp đề xuất.

IV. Phương pháp nghiên cứu của đề tài

Cách tiếp cận

Đề tài ban đầu sẽ nghiên cứu một số các phương pháp RGTT theo hướng tiếp cận tập thô và tập mờ nhằm tìm ra các ưu nhược điểm của mỗi phương pháp. Tiếp theo, đề tài sẽ đề xuất một số thuật toán gia tăng theo hướng tiếp cận tập mờ sử dụng lát cắt α có khả năng cải thiện hiệu năng phân lớp trên các bộ dữ liệu có tính nhiễu và thời gian xử lý trong trường hợp BQĐ thêm và loại bỏ TĐT. Cuối cùng, đề tài cũng làm rõ những ưu điểm của những phương pháp đề xuất thông qua quá trình phân tích và đánh giá các kết quả thực nghiệm khi so sánh với các phương pháp khác nhau trên các bộ dữ liệu tiêu chuẩn.

Các phương pháp nghiên cứu

- Nghiên cứu lý thuyết:

+ Nghiên cứu từ tổng quan tới chuyên sâu các lý thuyết nền tảng để từ đó tiếp cận đến những lý thuyết nâng cao.

+ Thu thập, tổng hợp, đánh giá và rút ra các kết luận cũng như hướng phát triển trên các kết quả đã được công bố về RGTT trên BQĐ.

+ Đề xuất, cải tiến và chứng minh các định nghĩa, mệnh đề sử dụng cho các phương pháp đề xuất một cách chặt chẽ.

- Nghiên cứu thực nghiệm:

+ Cài đặt thuật toán trên các bộ dữ liệu có độ tin cậy cao với kích thước từ trung bình đến lớn nhằm đánh giá và so sánh kết quả đã được công bố trên các tạp chí chuyên ngành có uy tín.

+ Áp dụng kết quả đạt được để xây dựng chương trình có tính ứng dụng cao.

V. Kết cấu của nội dung nghiên cứu

Đề án gồm:

- Chương 1: Tổng quan về lý thuyết tập thô, tập thô mờ và các ứng dụng trong bài toán rút gọn thuộc tính

- Chương 2: Lý thuyết tập mờ mức α và một số thuật toán gia tăng rút gọn thuộc tính

- Chương 3:Kết quả thực nghiệm thông qua quá trình phân tích, đánh giá và so sánh với các thuật toán.

Qua đó, sẽ đưa ra một số thảo luận và hướng nghiên cứu tiếp theo trong tương lai.

CHƯƠNG 1. TỔNG QUAN VỀ LÝ THUYẾT TẬP THÔ, TẬP THÔ MỜ VÀ CÁC ỨNG DỤNG TRONG BÀI TOÁN RÚT GỌN THUỘC TÍNH

1.1. LÝ THUYẾT TẬP THÔ, TẬP THÔ MỜ

1.1.1. Khái niệm cơ bản về tập thô

Vào đầu những năm 1980, nhà logic học Zdzisaw Pawlak đưa ra lý thuyết tập thô [1] và qua sự phát triển cũng như chứng minh trên một nền tảng toán học vững chắc, nó đã được coi là công cụ hiệu quả để giải quyết các bài toán về mô tả sự phụ thuộc giữa các thuộc tính, đánh giá độ quan trọng của các thuộc tính, phát hiện luật thu được và nhận dạng. Cho tới nay đã có rất nhiều hướng tiếp cận dựa trên lý thuyết tập thô được áp dụng thành công trong lĩnh vực khai phá dữ liệu và máy học như sinh luật quyết định hay trích chọn đặc trưng. Dựa trên sự phát triển của lý thuyết tập thô truyền thống mà các mô hình tập thô mở rộng ngày càng được ứng dụng rộng rãi để giải quyết các bài toán phân tích, khai phá dữ liệu, đặc biệt là các bài toán trên một khối lượng dữ liệu lớn, chứa đựng các thông tin mơ hồ, không chắc chắn mà điển hình là các hệ thông tin đầy đủ (Information System - IS) hay các hệ thông tin không đầy đủ (Incomplete Information System - IIS). Hệ thông tin giúp ích rất lớn cho việc lưu trữ và xử lý thông tin. Tuy nhiên, vì một lý do nào đó trong quá trình cập nhật mà thông tin lưu trữ có các thuộc tính dư thừa và tạo ra sự khó khăn trong việc khai phá tri thức.

Hệ thông tin là công cụ biểu diễn tri thức dưới dạng một bảng dữ liệu gồm p cột ứng với p thuộc tính và n hàng ứng với n đối tượng. Một cách hình thức, hệ thông tin được định nghĩa như sau:

Định nghĩa 1. *Hệ thông tin là một bộ tứ được biểu diễn dưới dạng* $IS = (U, A, V, f)$, *trong đó U là tập hữu hạn, khác rỗng các đối tượng; A là tập hữu hạn, khác rỗng các thuộc tính;* $V = \cup V_a$ *với* V_a *là tập giá trị của thuộc tính* $a \in A$; $f: U \times A \to V_a$ *là hàm thông tin,* $\forall a \in A, u \in U, f(u, a) \in V_a$.

Để đơn giản, với mọi $a \in A, u \in U$, ta ký hiệu giá trị thuộc tính a tại đối tượng u là $a(u)$ thay vì $f(u, a)$. Nếu $B = \{b_1, b_2, \dots, b_k\} \subseteq A$ là một tập con

các thuộc tính thì ta ký hiệu bộ các giá trị $b_i(u)$ bởi $B(u)$. Như vậy, nếu u và v là hai đối tượng thì ta viết $B(u) = B(v)$ nếu $b_i(u) = b_i(v)$ với mọi $i = 1, \dots, k$.

Xét một hệ thông tin $IS = (U, A, V, f)$, nếu tồn tại $u \in U$ và $a \in A$ sao cho $a(u)$ thiếu giá trị (missing value) thì IS được gọi là hệ thông tin không đầy đủ, ngược lại IS được gọi là hệ thông tin đầy đủ. Mỗi tập con các thuộc tính $B \subseteq A$ xác định một quan hệ hai ngôi trên U, ký hiệu là R_B và được xác định bởi:

$$R_B = \{(u, v) \in U \times U | \forall a \in B, a(u) = a(v)\} \quad (1.1)$$

R_B là quan hệ $B-$không phân biệt được. Rõ ràng, R_B là một quan hệ tương đương trên U. Nếu $(u, v) \in R_B$ thì hai đối tượng u và v không phân biệt được bởi các thuộc tính trong B. Quan hệ tương đương R_B sẽ xác định một phân hoạch trên U, ký hiệu là U/R_B hay để đơn giản là U/B. Mỗi phần tử của phân hoạch U/B được gọi là một lớp tương đương chứa đối tượng $u \in U$ và được ký hiệu là $[u]_B$, khi đó:

$$[u]_B = \{v \in U | (u, v) \in R_B\} \quad (1.2)$$

Định nghĩa 2. *Cho hệ thông tin $IS = (U, A, V, f)$ và $P, Q \subseteq A$, khi đó ta có:*

1. Phân hoạch U/P và U/Q được gọi là như nhau (viết là U/P = U/Q), khi và chỉ khi $\forall u \in U, [u]_P = [u]_Q$.

2. Phân hạch U/P mịn hơn phân hoạch U/Q (viết là $U/P \preccurlyeq U/Q$) khi và chỉ chi $\forall u \in U, [u]_P \subseteq [u]_Q$.

Xét hệ thông tin $IS = (U, A, V, f)$ và TĐT $X \in U$. Với một tập thuộc tính $B \subseteq A$ cho trước sẽ xác định được các lớp tương đương của phân hoạch U/B. Khi đó, một TĐT X cũng có thể được biểu diễn thông qua lớp tương đương này. Trong lý thuyết tập thô, để biểu diễn X thông qua các lớp tương đương của $X \subseteq U$, người ta xấp xỉ X bởi hợp của một số hữu hạn các lớp tương đương trong U/B. Có hai cách xấp xỉ TĐT X thông qua tập thuộc tính B, được gọi là B-xấp xỉ dưới và B-xấp xỉ trên của X, ký hiệu lần lượt là $\underline{B}X$ và $\overline{B}X$, được xác định như sau:

$$\underline{B}X = \{u \in U | [u]_B \subseteq X\} \quad (1.3)$$

$$\overline{B}X = \{u \in U | [u]_B \cap X \neq \phi\} \quad (1.4)$$

Tập $\underline{B}X$ bao gồm tất cả cá phần tử của U chắc chắn thuộc vào X, còn tập $\overline{B}X$ bao gồm các phần tử của U có thể thuộc vào X dựa trên tập thuộc tính B. Từ hai tập xấp xỉ nêu trên, ta định nghĩa các tập B-miền biên của X và B-miền ngoài của X, lần lượt theo hai công thức dưới đây:

$$NB(X) = \overline{B}X \backslash \underline{B}X \quad (1.5)$$

$$MB(X) = U \backslash \overline{B}X \quad (1.6)$$

B-miền biên của X là tập chứa các đối tượng có thể thuộc hoặc không thuộc X còn B- miền ngoài của X là tập chứa các đối tượng chắc chắn không thuộc X. Trong trường hợp $\underline{B}X=\emptyset$ thì X được gọi là tập chính xác, ngược lại X được gọi là tập thô. Với $B, D \subseteq A$, ta gọi B-miền dương của D là tập được xác định như sau:

$$POS_B(D) = \bigcup_{X \in \frac{U}{D}} (B\underline{X}) \quad (1.7)$$

Rõ ràng, $POS_B(D)$ là tập tất cả các đối tượng u sao cho với mọi đối tượng $v \in U$ mà $u(B) = v(B)$ ta đều có $u(D) = v(D)$. Nói cách khác, $POS_B(D) = \{u \in U | [u]_B \subseteq [u]_D\}$.

Trong nhiều ứng dụng, một loại hệ thông tin đặc biệt đóng vai trò quan trọng, được gọi là BQĐ. BQĐ là một hệ thống thông tin DS với tập thuộc tính A được phân chia thành hai phần không giao nhau: C và D. C được gọi là tập thuộc tính điều kiện và D là tập thuộc tính quyết định, để đơn giản chúng tôi ký hiệu BQĐ là $DS = (U, C \cup D)$ với $C \cap D \neq \phi$. Với mọi $d \in D$, $d(u)$ đầy đủ giá trị, nếu tồn tại $u \in U$ và $c \in C$sao cho $c(u)$ thiếu giá trị thì DS được gọi là BQĐ không đầy đủ, trái lại DS được gọi là BQĐ đầy đủ. Trong phạm vi nghiên cứu này, chúng tôi chỉ xét tới BQĐ đầy đủ.

Ví dụ 1. Cho BQĐ $DS = (U, C \cup D)$, trong đó $U = \{u_1, u_2, u_3, u_4\}$và $C = \{c_1, c_2, c_3, c_4\}$.

Bảng 1.1: ***Bảng quyết định đầy đủ***

U	c_1	c_2	c_3	c_4	D
u_1	0.8	0.6	1	0	0
u_2	0.8	0	0.2	0.8	1
U_3	0.6	0.8	0.6	0.4	0
U_4	0	0.6	0	1	1

1.1.2. Khái niệm cơ bản về tập thô mờ

Lý thuyết TTM (fuzzy rough set) do Dubois và các cộng sự [2-3] đề xuất là sự kết hợp của lý thuyết tập thô và lý thuyết tập mờ nhằm xấp xỉ các tập mờ dựa trên một QHTĐM (fuzzy equivalence relation) được xác định trên miền giá trị thuộc tính. Về bản chất, các QHTĐM được mở rộng từ các quan hệ tương đương mà báo cáo đã trình bày trong phần trước.

Định nghĩa 3. *Cho BQĐ* $DS = (U, C \cup D)$*, một quan hệ* $\tilde{R}$ *xác định trên miền giá trị thuộc tính được gọi là QHTĐM nếu thỏa mãn các điều kiện sau với mọi* $u, v, t \in U$.

1. *Tính phản xạ:* $\tilde{R}(u,u) = 1$.
2. *Tính đối xứng:* $\tilde{R}(u,v) = \tilde{R}(v,u)$.
3. *Tính bắc cầu sup-min:* $\tilde{R}(u,v) \geq \sup_{t \in U} \{min(\tilde{R}(u,t), \tilde{R}(t,v))\}$.

Mệnh đề 1. *Cho BQĐ* $DS = (U, C \cup D)$*và một QHTĐM* $\tilde{R}$*. Ký hiệu* $\tilde{R}_P, \tilde{R}_Q$ *tương ứng là các quan hệ* $\tilde{R}$ *xác định trên tập thuộc tính* $P, Q \subseteq C$*. Khi đó, với mọi* $u, v \in U$*, ta có:*

1. $\tilde{R}_P = \tilde{R}_Q \Leftrightarrow \tilde{R}_P(u,v) = \tilde{R}_Q(u,v)$
2. $\tilde{R}_{P \cap Q} = \tilde{R}_P \cup \tilde{R}_Q = max\{\tilde{R}_P(u,v), \tilde{R}_Q(u,v)\}$.
3. $\tilde{R}_{P \cup Q} = \tilde{R}_P \cap \tilde{R}_Q = min\{\tilde{R}_P(u,v), \tilde{R}_Q(u,v)\}$.
4. $\tilde{R}_P \subseteq \tilde{R}_Q \Leftrightarrow \tilde{R}_P(u,v) \leq \tilde{R}_Q(u,v)$.

Định nghĩa 4. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$*và* $\tilde{R}_P$ *là QHTĐM xác định trên tập thuộc tính* $P \subseteq C$*. Khi đó, ma trận tương đương mờ biểu diễn* $\tilde{R}_P$*, ký hiệu là* $M(\tilde{R}_P) = [p_{ij}]_{n\times m}$*, được định nghĩa như sau:*

$$M(\tilde{R}_P) = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix}$$

với $p_{ij} = \tilde{R}_p(u_i, u_j)$ *là giá trị quan hệ giữa hai đối tượng* u_i *và* u_j *trên tập thuộc tính* P,$p_{ij} \in [0,1]$, $u_i, u_j \in U$,$1 \le i, j \le n$.

Như vậy, ta có thể nhận thấy rằng giá trị của các phần tử trong ma trận tương đương mờ $M(\tilde{R}_P)$ phụ thuộc vào QHTĐM $\tilde{R}_P$ được chọn. Mặt khác, ma trận tương đương mờ là nền tảng để xây dựng các độ đo được sử dụng để giải quyết bài toán RGTT trong BQĐ mà báo cáo sẽ làm rõ hơn trong các phần tiếp theo.

Mệnh đề 2. *Cho BQĐ* $DS = (U, C \cup D)$ *và* $P, Q \subseteq C$*. Giả sử* $M(\tilde{R}_P) = [pi_j]_{n\times m}, M(\tilde{R}_Q) = [q_{ij}]_{n\times m}$ *tương ứng là các ma trận tương đương mờ của quan hệ* $\tilde{R}_P$ *và* $\tilde{R}_Q$*, khi đó ma trận tương đương mờ trên tập thuộc tính* $S = P \cup Q$ *là:*

$$M(\tilde{R}_P) = M\left(\tilde{R}_{P\cup Q}\right) = [s_{ij}]_{n\times m} \tag{1.8}$$

trong đó, $s_{ij} = min(p_{ij}, q_{ij})$

Chứng minh: Theo mệnh đề 1, ta có $\tilde{R}_P = \underset{a\in P}{\cap} \tilde{R}_{\{a\}}$ và $\tilde{R}_{P\cup Q} = \tilde{R}_P \cap \tilde{R}_Q$, có nghĩa là với mọi đối tượng $u, v \in U$ thì $\tilde{R}_{P\cup Q}(u, v) = min(\tilde{R}_P(u, v), \tilde{R}_Q(u, v))$. Từ đó, ta có $M(\tilde{R}_s) = M(\tilde{R}_{P\cup Q}) = [s_{ij}]_{n\times m}$ với $s_{ij} = min(p_{ij}, q_{ij})$.

Định nghĩa 5. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $P, Q \subseteq C$, $U = \{u_1, u_2, \dots, u_n\}$ *và* $\tilde{R}_P$ *là QHTĐM trên tập thuộc tính P. Khi đó, phân hoạch mờ trên U sinh bởi* $\tilde{R}_P$*, ký hiệu là* $\tilde{\Upsilon}_P$*, được xác định như sau:*

$$\tilde{\Upsilon}_P = \frac{U}{\tilde{R}_P} = \{[\widetilde{u_i}]_P\}_{i=1}^{n} = \{[\widetilde{u_1}]_P, [\widetilde{u_2}]_P, \ldots, [\widetilde{u_n}]_P\} \quad (1.9)$$

trong đó, $[\widetilde{u_i}]_P = \{\frac{p_{i1}}{u_1}, \frac{p_{i2}}{u_2}, \ldots, \frac{p_{in}}{u_n}\}$ *là một tập mờ đóng vai trò là một lớp tương đương mờ của đối tượng* $u_i \in U$.

Với lớp tương đương mờ $[\widetilde{u_i}]_P$, hàm thuộc của tất cả các đối tượng $u_i \in U$ được xác định bởi $\mu_{[\widetilde{u_i}]_P}(u_j) = \mu_{\tilde{R}_P}(u_i, u_j) = \tilde{R}_P(u_i, u_j)$ và lực lượng của lớp tương đương mờ $[\widetilde{u_i}]_P$ được tính bởi $|[\widetilde{u_i}]_P| = \sum_{j=1}^{n} p_{ij}$.

Ví dụ 2. *Xét BQĐ trong ví dụ 1, với một QHTĐM trên mỗi thuộc tính* $a \in C$ *được xác định bởi công thức* $\tilde{R}_{\{a\}}(u, v) = 1 - |a(u) - a(v)|$, *khi đó theo định nghĩa 4, ma trận tương đương mờ của thuộc tính* c_1 *là:*

$$M\left(\tilde{R}_{\{c_1\}}\right) = \begin{bmatrix} 1.0 & 1.0 & 0.8 & 0.2 \\ 1.0 & 1.0 & 0.8 & 0.2 \\ 0.8 & 0.8 & 1.0 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1.0 \end{bmatrix}$$

Theo Định nghĩa 5, $[\widetilde{u_1}]_{\{c_1\}} = \left\{\frac{1}{u_1}, \frac{1}{u_2}, \frac{0.8}{u_3}, \frac{0.2}{u_4}\right\}$ *là lớp tương đương mờ của đối tượng* u_1 *và lực lượng của* $[\widetilde{u_1}]_{\{c_1\}} = 1 + 1 + 0.8 + 0.2 = 3$. *Phân hoạch mờ của quan hệ mờ* $\tilde{R}_{\{c_1\}}$ *là* $\tilde{\Upsilon}_{\{c_1\}} = \{[\widetilde{u_1}]_{\{c_1\}}, [\widetilde{u_2}]_{\{c_1\}}, [\widetilde{u_3}]_{\{c_1\}}, [\widetilde{u_4}]_{\{c_1\}}\}$.

Định nghĩa 6. *Cho* $\tilde{X}$ *là một tập mờ trên* U *và* $\tilde{R}_P$ *là một QHTĐM trên tập thuộc tính* $P \subseteq C$. *Khi đó, tập xấp xỉ dưới mờ* $\underline{P}\tilde{X}$ *và tập xấp xỉ trên mờ* $\overline{P}\tilde{X}$ *của* $\tilde{X}$ *là các tập mờ và có hàm thuộc của các đối tượng* $u \in U$ *được xác định như sau:*

$$\mu_{\underline{P}\tilde{X}}(u) = \inf_{v \in U} max\left(1 - \mu_{[\widetilde{u}]_P}(v), \mu_{\tilde{X}}(v)\right) \quad (1.10)$$

$$\mu_{\overline{P}\tilde{X}}(u) = \sup_{v \in U} min\left(\mu_{[\widetilde{u}]_P}(v), \mu_{\tilde{X}}(v)\right) \quad (1.11)$$

Cặp $\left(\underline{P}\tilde{X} = \overline{P}\tilde{X}\right)$ được gọi là TTM. Dễ thấy, một tập rõ $X \in U$ cũng được biểu diễn tri thức bởi hai công thức trên khi coi nó là một tập mờ với hàm thuộc $\mu_X(v) = 1$ với $v \in X$ và $\mu_X(v) = 0$ với $v \notin X$. Mô hình TTM có thể xem là việc sử dụng quan hệ tương tự để xấp xỉ tập mờ (hoặc tập rõ) bằng tập mờ xấp

xỉ dưới và tập mờ xấp xỉ trên. Trong lý thuyết tập thô truyền thống, khái niệm miền dương được định nghĩa là hợp của tất cả các tập xấp xỉ dưới. Trong lý thuyết TTM, miền dương mờ được định nghĩa như sau.

Định nghĩa 7. *Cho BQĐ* $DS = (U, C \cup D)$, $\tilde{R}_P$ *và* $\tilde{R}_D$ *tương ứng là hai QHTĐM xác định trên* $P \subseteq C$ *và D. Khi đó, miền dương mờ của tập thuộc tính điều kiện D với tập thuộc tính P, được ký hiệu là* $\widetilde{POS_P}(D)$*và có hàm thuộc của mỗi đối tượng* $u \in U$ *được xác định như sau:*

$$\mu_{\widetilde{POS_P}(D)}(u) = \sup_{u \in \underline{\underline{U}}_{\tilde{R}_D}} \mu_{\underline{P}\tilde{X}}(u) \tag{1.12}$$

Dễ thấy $\widetilde{POS_P}(D)$ là một tập mờ và được mở rộng từ khái niệm miền dương mờ từ lý thuyết tập thô truyền thống. Dựa trên khái niệm này, chúng tôi định nghĩa độ phụ thuộc của một tập con thuộc tính như sau.

Định nghĩa 8. *Cho BQĐ* $DS = (U, C \cup D)$, $\tilde{R}_P$*và* $\tilde{R}_D$ *tương ứng là hai QHTĐM xác định trên* $P \subseteq C$ *và D. Độ phụ thuộc của tập thuộc tính P với tập thuộc tính quyết định D được định nghĩa như sau:*

$$\gamma_P(D) = \frac{|\widetilde{POS_P}(D)|}{|U|} = \frac{\sum_{u \in U} \mu_{\widetilde{POS_P}(D)}(u)}{|U|} \tag{1.13}$$

1.2. MỘT SỐ PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH DỰA TRÊN LÝ THUYẾT TẬP THÔ VÀ MỞ RỘNG

RGTT là quá trình giảm hay lược bỏ các đặc trưng/thuộc tính trong tập dữ liệu nguyên thủy. Mục tiêu của việc RGTT là tạo ra một tập dữ liệu có kích thước nhỏ hơn mà vẫn giữ được các thông tin cần thiết và mô tả được những đặc trưng cốt lõi của dữ liệu gốc. Quá trình này thường được thực hiện để tăng tính hiệu quả của việc xử lý và phân tích dữ liệu, giảm chi phí tính toán và làm cho dữ liệu dễ dàng quản lý hơn. Các kỹ thuật RGTT chia làm hai nhóm: Lựa chọn thuộc tính (LCTT) và biến đổi thuộc tính (BDTT). LCTT là trích chọn một tập con tối ưu (theo một nghĩa nào đó) từ tập thuộc tính nguyên thủy. BDTT

là thực hiện việc chuyển đổi các thuộc tính ban đầu thành một tập các thuộc tính mới với kích thước ít hơn sao cho bảo toàn được thông tin ở mức tối đa.

Các công trình nghiên cứu về RGTT thường tập trung vào nghiên cứu các kỹ thuật LCTT. LCTT là quá trình chọn ra một tập con có kích thước |B| từ tập gốc chứa |C| thuộc tính ($B \subseteq C$), sao cho không gian thuộc tính được thu gọn một cách tối ưu dựa trên một tiêu chuẩn cụ thể. Việc tìm ra tập con thuộc tính tối ưu thường là một vấn đề khó; thực tế, nó thuộc vào lớp bài toán NP-khó. Thông thường, một thuật toán lựa chọn thuộc tính bao gồm bốn khâu cơ bản.

(1) Khởi tạo tập con;

(2) Phân tích tập con;

(3) Xét điều kiện dừng;

(4) Đánh giá kết quả.

Tạo lập tập con thuộc tính là quá trình liên tục tìm kiếm nhằm tạo ra các tập con để đánh giá và lựa chọn. Giả sử tập dữ liệu ban đầu chứa |C| thuộc tính. Với |C| thuộc tính này, tổng số tập con có thể được tạo ra là 2|C|. Do đó, việc tìm ra tập con tối ưu từ tất cả các tập con này là rất khó khăn. Một phương pháp phổ biến để tìm kiếm tập con thuộc tính tối ưu là tạo ra từng tập con để so sánh. Mỗi tập con được tạo ra sẽ được đánh giá dựa trên một tiêu chuẩn nhất định và so sánh với tập con tốt nhất đã được chọn trước đó. Nếu tập con mới này cải thiện, nó sẽ thay thế tập con cũ. Quá trình tìm kiếm tập con thuộc tính tối ưu sẽ dừng khi một trong bốn điều kiện sau xảy ra:

(1) Đã thu được số thuộc tính dựa trên 1 tiêu chuẩn.

(2) Số bước lặp được định nghĩa trong quá trình kết thúc.

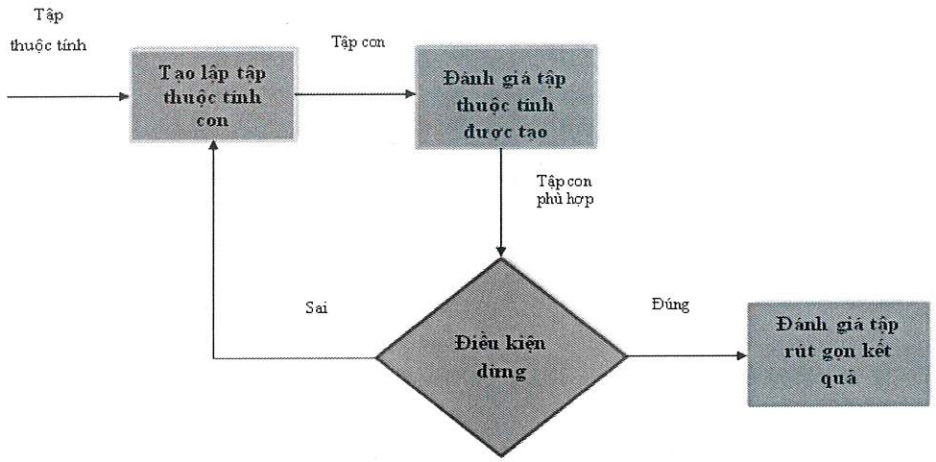
(3) Việc bổ sung vào hay lược bỏ một thuộc tính nào đó không làm cho kết quả tốt hơn.

(4) Đã thu được tập con tốt nhất theo tiêu chuẩn đánh giá.

Cuối cùng, tập con tốt nhất phải được xác minh thông qua việc thực hiện các phép kiểm định, so sánh kết quả khai phá với tập thuộc tính "tốt nhất" này

và tập thuộc tính ban đầu trên các tập dữ liệu khác nhau. Quá trình lựa chọn thuộc tính được biểu diễn như hình sau (Hình 1.1).

Hiện nay, có hai phương pháp chính để tiếp cận bài toán lựa chọn thuộc tính: Lọc (filter) và Đóng gói (wrapper), mỗi phương pháp này đều có mục tiêu riêng về việc giảm số lượng thuộc tính hoặc nâng cao độ chính xác của mô hình phân loại. Phương pháp lọc thực hiện việc lựa chọn thuộc tính độc lập với các thuật toán khai phá sử dụng sau này. Các thuộc tính được chọn dựa trên độ quan trọng của chúng trong việc mô tả dữ liệu. Phương pháp này có ưu điểm là thời gian tính toán nhanh, nhưng nhược điểm là không sử dụng thông tin nhãn lớp của các bộ dữ liệu, do đó độ chính xác không cao.Ngược lại, phương pháp đóng gói thực hiện bằng cách áp dụng ngay kỹ thuật khai phá cụ thể với TRG thuộc tính, độ chính xác của kết quả được sử dụng làm tiêu chuẩn để lựa chọn các tập con thuộc tính.



Hình 1.1: ***Quy trình RGTT***

1.2.1. Phương pháp rút gọn thuộc tính theo tiếp cận tập thô

Cho đến nay có rất nhiều các phương pháp RGTT trong BQĐ đầy đủ theo tiếp cận lý thuyết tập thô truyền thống, các phương pháp điển hình được trình bày như sau:

- Phương pháp RGTT dựa trên miền dương:

Kể từ khi Pawlak đưa ra định nghĩa TRG dựa trên miền dương, các công trình nghiên cứu đã xây dựng thuật toán tính miền dương, Dựa trên điều đó, ta phát triển một thuật toán để tìm TRG dựa trên miền dương. Cụ thể, một rút gọn được định nghĩa như sau:

Định nghĩa 9. *Cho BQĐ* $DS = (U, C \cup D)$*, một tập* $B \subseteq C$ *được gọi là một TRG của C dựa trên miền dương nếu thỏa mãn:*

1. $POS_B(D) = POS_C(D)$
2. $\forall b \in B, POS_{B\setminus\{b\}}(D) \neq POS_B(D)$

Định nghĩa 10. *Cho BQĐ* $DS = (U, C \cup D)$ *và một tập* $B \subseteq C$*. Khi đó độ quan trọng của thuộc tính* $b \in C$ *được tính theo công thức sau:*

$$SIG(b, B) = \gamma_B(D) - \gamma_{B\setminus\{b\}}(D) \tag{1.14}$$

Rõ ràng, độ cần thiết của thuộc tính theo Định nghĩa 10 có tính đơn điệu. Sự thay đổi trong hàm phụ thuộc càng cao thì thuộc tính càng quan trọng. Do đó, khi xây dựng thuộc tính, các thuật toán sẽ sử dụng định nghĩa này để xây dựng một chuỗi các thuộc tính ứng viên cho TRG. Dựa trên Định nghĩa 9 và 10, Hoa và các cộng sự tại [4] đã sử dụng phương pháp sắp xếp nhanh (Quicksort) để sắp xếp các đối tượng theo tính phù hợp và xây dựng thuật toán tính miền dương. Xu và các cộng sự trong [5] sử dụng phương pháp sắp xếp theo cơ số (Radix-sort) để xây dựng thuật toán tính miền dương. Dựa trên tính đơn điệu thông qua tính cần thiết của dữ liệu được trình bày trong Định nghĩa 10 và hai tính chất của TRG từ Định nghĩa 9, Shu và các cộng sự trong [6] đã xây dựng thuật toán lọc GFS để tìm kiếm các thuộc tính quan trọng trên BQĐ. Các bước của thuật toán GFS được trình bày trong mã giả 1. Dựa trên thuật

toán GFS, các tác giả trong [6] đã mở rộng công thức tính toán hàm độc lập và đề xuất thuật toán gia tăng là IFSA sử dụng khi bảng quyết đỉnh bổ sung TĐT và IFSD sử dụng khi BQĐ loại bỏ TĐT. Các kết quả thực nghiệm đã cho thấy, các phương pháp đề xuất có hiệu quả cao hơn so với các phương pháp trong [7, 8]. Báo cáo này sẽ trình bày chi tiết các bước của thuật toán GFS, IFSA và IFSD để từ đó thấy được những ưu, nhược điểm của phương pháp khi dựa trên độ đo miền dương truyền thống. Báo cáo cũng sử dụng các thuật toán này làm cơ sở để so sánh với các thuật toán đề xuất được trình bày trong các phần tiếp theo.

Thuật toán GFS bao gồm ba giai đoạn chính. Giai đoạn thứ nhất sẽ loại bỏ đi các thuộc tính có độ quan trọng bằng 0 trên tập thuộc tính điều kiện C. Mục đích của giải đoạn này là giảm thiểu không gian tìm kiếm cho các bước sau đó của thuật toán. Giai đoạn thứ hai thuật toán thuật toán sẽ lọc tiếp trên các thuộc tính tìm được ở giai đoạn 1 để chọn các thuộc tính quan trọng nhất. Đây cũng là giai đoạn chủ chốt của thuật toán khi độ cần thiết của mỗi thuộc tính sẽ được đánh giá trên TRG thu được từ bước trước đó. Nói một cách khác, thuật toán sẽ kiểm chứng xem mức ảnh hưởng của thuộc tính được lựa chọn tiếp theo đối với TRG thu được. Giai đoạn cuối cùng sẽ tiếp tục xáo bỏ các thuộc tính không quan trọng để thu được một TRG tối ưu.

Algorithm 1: The attribute reduction algorithm GFS

Input: A decision table $DS = (U, C \cup D)$

Output: One reduct B

1. initialize: $B := \phi$
2. compute the new dependency function $\gamma_C(D)$
3. **for** $a \in C$ **do**
4. | compute $SIG(a, C)$
5. | **if** $SIG(a, C) > 0$ **then** $B := B \cup \{a\}$
6. **end for**

7. let $P = B$
8. **while** $\gamma_P(D) \neq \gamma_C(D)$**do**
9. | **for** $a \in C \backslash P$ **do**
10. | select $P := P \cup \{a_0\}$, where $a_0 = argmax\{SIG(a, P \cup \{a\})\}$
11. **end while**
12. **for** $a \in P$**do**
13. | compute $SIG(a, P)$
14. | **if** $SIG(a, P) = 0$, **then** $P := P \backslash \{a\}$
15. **end for**
16. set $B := P$
17. **Return** B

Từ Định nghĩa 8, nhóm nghiên cứu trong [6] tiếp tục mở rộng công thức tính độ phụ thuộc cho cả hai trường hợp gia tăng và loại bỏ TĐT.

Định lý 1. *Cho BQĐ* $DS = (U, C \cup D)$, *tập thuộc tính* $B \subseteq C$, $\frac{U}{B} = \{X_1, X_2, \ldots, X_m\}$ *và* $\frac{U}{B} = \{Y_1, Y_2, \ldots, Y_m\}$. *Giả sử rằng giá trị B-miền dương của D trên TĐT U là* $POS_B^U(D)$, *TĐT bổ sung là* U_{ad} *có hai tập phân hoạch* $\frac{U_{ad}}{B} = \{M_1, M_2, \ldots, M_{m'}\}$ *và* $\frac{U_{ad}}{D} = \{Z_1, Z_2, \ldots, Z_{n'}\}$, *cặp phân hoạch thuộc tính B và D là* $U \cup \frac{U_{ad}}{B} = \{X_1', X_2', \ldots, X_l', X_{l+1}', X_{l+2}', \ldots, X_m, M_{l+1}, M_{l+2}, \ldots, M_{m'}\}$ *và* $U \cup \frac{U_{ad}}{D} = \{Y_1', Y_2', \ldots, Y_k', Y_{k+1}', Y_{k+2}', \ldots, Y_n, Y_{k+1}, Y_{k+2}, \ldots, Z_{n'}\}$ *trên toàn bộ bảng. Khi đó, độ phụ thuộc mới của tập thuộc tính B theo D trên toàn bộ bảng được tính như sau:*

$$\gamma_B^{U \cup U_{ad}}(D) = \frac{POS_B^U(D)}{|U \cup U_{ad}|} + \frac{POS_B^{U_{ad}}(D)}{|U \cup U_{ad}|} - \frac{\left|\left\{x \in X_i' \middle| \left|\frac{X_i'}{D}\right| \neq 1\right\}\right|}{|U \cup U_{ad}|} (1 \leq i \leq l) \qquad (1.15)$$

Việc tính toán hàm phụ thuộc theo Định lý 1 đóng vai trò quan trọng trong các thuật toán RGTT khi ảnh hưởng trực tiếp đến hiệu quả của việc lựa

chọn tập con đặc trưng. Từ Định lý 1, thuật toán GFS được phát triển thành thuật toán IFSA được sử dụng khi BQĐ có sự bổ sung của TĐT. Thời gian tính toán TRG của thuật toán IFSA sẽ được giảm thiểu đáng kể.

Algorithm 2: Incremental attribute reduction when adding the objects set (IFSA)

Input: $DS = (U, C \cup D)$ the reduct B_U on U and the set of adding objects U_{ad}.

Output: A new reduct B' on $U \cup U_{ad}$.

1. initialize: $P := B'$ and $U' = U \cup U_{ad}$
2. compute the partitions of U on C and P respectively, $\frac{U}{D} = \{X_1, X_2, \dots, X_m\}$ and $\frac{U}{B} = \{X_1, X_2, \dots, X_s\}$.
3. compute the partitions of U_{ad} on condition attribute set C and P respectively, $\frac{U_{ad}}{C} = \{M_1, M_2, \dots, M_{m'}\}$ and $\frac{U_{ad}}{P} = \{M_1, M_2, \dots, M_{s'}\}$.
4. compute the partitions of the new object set U' on C *and* P respectively, $\frac{U'}{C} = \{X'_1, X'_2, \dots, X'_{l'}, X'_{l'+1}, X'_{l'+2}, \dots, X_m, M_{l'+1}, M_{l'+2}, \dots, M_{m'}\}$and $\frac{U'}{P} = \{X'_1, X'_2, \dots, X'_l, X'_{l+1}, X'_{l+2}, \dots, X_m, M_{l+1}, M_{l+2}, \dots, M_{s'}\}$.
5. compute the new dependency function $\gamma_P^{U'}(D)$ and $\gamma_C^{U'}(D)$ by Theorem 1.
6. **if** $\gamma_P^{U'}(D) = \gamma_C^{U'}(D)$ **then** go to step 12; **else** go to step 7.
7. **for** $\forall c \in C \backslash P$, construct a descending sequence by $SIG(c, P)$, and record the results by $\{c'_1, c'_2, \dots, c'_{|C \backslash P|}\}$.
8. **while** $\gamma_P(D) \neq \gamma_C(D)$**do**
9. | **for** c in $C \backslash P$
10. | Select $P := P \cup \{c\}$and compute $\gamma_P^{U'}(D)$
11. **end while**
12. **for** each attribute $p \in P$ **do**
13. | compute $SIG(p, P)$

14. **if** $SIG(p,P) = 0$, **then** $P := P\backslash\{p\}$
15. **end for**
16. $B' = P$ and **return** B'

Khi TĐT được bổ sung vào BQĐ, quy trình chi tiết của Thuật toán IFSA được trình bày cụ thể như sau. Các bước 2-5 là tính toán phân hoạch và cập nhật hàm phụ thuộc theo công thức gia tăng dựa trên Định lý 1; bước 6 kiểm xem hàm phụ thuộc mới của tập con thuộc tính giai đoạn trước đó với TĐT cập nhật có bằng với hàm phụ thuộc trong toàn bộ tập thuộc tính điều kiện hay không (nếu bằng nhau thì giữ nguyên tập thuộc tính ban đầu). Các bước 7-11 là xây dựng trình tự giảm dần cho các thuộc tính còn lại và cập nhật TRG tăng dần. Bước 12-15 là xóa các thuộc tính dư thừa khỏi kết quả lựa chọn.

Cũng dựa trên Định nghĩa 8, các tác giả trong [6] mở rộng công thức gia tăng trên BQĐ trong trường hợp loại bỏ TĐT.

Định lý 2. *Cho BQĐ* $DS = (U, C \cup D)$, *tập thuộc tính* $B \subseteq C$, $\frac{U}{B} = \{X_1, X_2, \dots, X_m\}$, $\frac{U}{D} = \{Y_1, Y_2, \dots, Y_n\}$. *Giả sử rằng giá trị B-miền dương của D trên TĐT U là* $POS_B^U(D)$, *TĐT* U_{de} *là TĐT bị loại bỏ, cặp phân hoạch thuộc tính B và D trên TĐT* $U\backslash U_{de}$ *lần lượt là* $\frac{(U\backslash U_{de})}{B} = \{X_1', X_2', \dots, X_z', X_{z+1}, X_{z+2}, \dots, X_m\}$, $\frac{(U\backslash U_{de})}{D} = \{Y_1', Y_2', \dots, Y_s', Y_{s+1}, Y_{s+2}, \dots, Y_n\}$. *Khi đó, độ phụ thuộc mới của tập thuộc tính B theo D trên toàn bộ bảng được tính theo công thức sau:*

$$\gamma_B^{U\backslash U_{de}} = \frac{|POS_B^U(D)|}{|U\backslash U_{de}|} - \frac{|U_{de}|}{|U\backslash U_{de}|} + \frac{\left|\left\{x_i' \middle| \left|\frac{X_i'}{D}=1\right|\right\}\right|}{|U\backslash U_{de}|} (1 \le i \le z) \qquad (1.16)$$

Từ Định lý 2, giá trị của hàm phụ thuộc mới có thể giảm khi loại bỏ nhiều đối tượng trên BQĐ. Từ đó, [6] cũng trình bày thuật toán gia tăng trong trường hợp BQĐ loại bỏ TĐT IFSD. Từ thuật toán này, hiệu quả của việc lựa chọn đặc trưng được cải thiện từ hai khía cạnh:

(1) hàm phụ thuộc được cập nhật tăng dần theo Định lý 2;

(2) tập con thuộc tính được cập nhật dần dần theo từng vòng lặp.

Chúng tôi trình bày chi tiết các bước tiến hành của thuật toán IFSD như sau:

Algorithm 3 Incremental attribute reduction when deleting the objects set (IFSD)

Input: $DS = (U, C \cup D)$, the reduct B_U on U and the set of deleting objects U_{ad}.

Output: A new reduct B' on $U \backslash U_{de}$

1. initialize: $P := B'$ and $U' := U \backslash U_{de}$
2. compute the partitions of U on C and P respectively, $\frac{U}{C} = \{X_1, X_2, \ldots, X_m\}$ and $\frac{U}{P} = \{X_1, X_2, \ldots, X_s\}$.
3. compute the partitions of the new object set U' on C *and* P respectively, $\frac{U'}{C} = \{X_1', X_2', \ldots, X_{z'}', X_{z'+1}, X_{z'+2}, \ldots, X_m, M_{z'+1}, M_{z'+2}, \ldots, M_{m'}\}$ and $\frac{U'}{P} = \{X_1', X_2', \ldots, X_{z'}', X_{z+1}, X_{z+2}, \ldots, X_m\}$.
4. Compute the new dependency function $\gamma_P^{U'}(D)$ and $\gamma_C^{U'}(D)$ by Theorem 2.
5. **if** $\gamma_P^{U'}(D) = \gamma_C^{U'}(D)$**then** go to step 11; **else** go to step 6.
6. **for** $\forall c \in C \backslash P$, construct a descending sequence by $SIG(c, P)$, and record the results by $\{c_1', c_2', \ldots, c_{|C \backslash P|}'\}$.
7. **while** $\gamma_P(D) \neq \gamma_C(D)$**do**
8. | **for** c in $C \backslash P$
9. | | Select $P := P \cup \{c\}$and compute $\gamma_P^{U'}(D)$
10. **end while**
11. **for** each attribute $p \in P$ **do**
12. | compute $SIG(p, P)$

13. **if** $SIG(p,P) = 0$, **then** $P := P\backslash\{p\}$
14. **end for**
15. $B' = P$ and **return** B'

- Phương pháp rút gọn thuộc tính dựa trên entropy Shannon:

Giống như các phương pháp RGTT khác, để xây dựng phương pháp heuristic sử dụng entropy Shannon, cần tiến hành nghiên cứu các bước:

(1) Định nghĩa TRG dựa trên entropy Shannon;

(2) Định nghĩa độ quan trọng của thuộc tính sử dụng entropy Shannon.

Độ quan trọng của thuộc tính đặc trưng cho chất lượng phân lớp của thuộc tính và là tiêu chuẩn lựa chọn thuộc tính trong các bước của thuật toán heuristic tìm một TRG có chất lượng phân lớp tốt nhất.

Định nghĩa 11. *Cho BQĐ* $DS = (U, C \cup D)$ *và tập thuộc tính* $P \subseteq C$. *Giả sử rằng* $\frac{U}{P} = \{P_1, P_2, \dots, P_m\}$, *khi đó entropy Shannon của P được xác định bởi công thức:*

$$ES(P) = -\sum_{i=1}^{m} \frac{|P_i|}{|U|} log_2 \frac{|P_i|}{|U|} \quad (1.17)$$

Có thể thấy rằng, nếu $\frac{U}{P} = U$ thì $ES(P) = 0$ và đạt giá trị nhỏ nhất. Ngược lại, nếu $P_i = \{u_i\} \forall u_i \in U, i \in [1, |U|]$ thì $ES(P)$ đạt giá trị lớn nhất tại $log_2|U|$.

Định nghĩa 12. *Cho BQĐ* $DS = (U, C \cup D)$, *giả sử rằng* $\frac{U}{C} = \{C_1, C_2, \dots, C_m\}$ *và* $\frac{U}{D} = \{D_1, D_2, \dots, D_n\}$, *khi đó entropy Shannon có điều kiện của D khi đã biết C được định nghĩa bởi:*

$$ES(D|C) = -\sum_{i=1}^{m} \frac{|C_i|}{|U|} \sum_{j=1}^{n} \frac{|C_i \cap D_j|}{|C_i|} log \frac{|C_i \cap D_j|}{|C_i|} \quad (1.18)$$

Mệnh đề 3. *Cho BQĐ* $DS = (U, C \cup D)$. *Nếu* $Q \subseteq P \subseteq C$ *thì* $ES(D|Q) \geq ES(D|P)$.

Mệnh đề 3 nói lên tính phản đơn điệu của entropy Shannon có điều kiện, nghĩa là tập thuộc tính điều kiện Q càng nhỏ (phân hoạch sinh bởi Q càng thô) thì $ES(D|Q)$ càng lớn và ngược lại.

Định nghĩa 13. *Cho BQĐ* $DS = (U, C \cup D)$*, thuộc tính* $a \in C$ *được gọi là dư thừa trong DS dựa trên Entropy Shannon có điều kiện nếu* $ES(D|C) = ES(D|C\backslash\{a\})$*. Ngược lại, a gọi là thuộc tính cần thiết. Tập tất cả các thuộc tính cần thiết trong DS được gọi là tập lõi dựa trên entropy Shannon có điều kiện và ký hiệu là* $HCORE(C)$*.*

Định nghĩa 14. *Cho BQĐ* $DS = (U, C \cup D)$ *và tập thuộc tính* $B \subseteq C$*. Khi đó B được gọi là rút gọn của C dựa trên entropy Shannon có điều kiện, gọi tắt là TRG Entropy Shannon nếu:*

1. $ES(D|B) = ES(D|C)$.
2. $\forall b \in B, ES(D|B\backslash\{b\}) \neq ES(D|C)$.

Định nghĩa 15. *Cho BQĐ* $DS = (U, C \cup D)$ *và tập thuộc tính* $B \subseteq C$*,* $b \in C\backslash B$*. Độ quan trọng của thuộc tính b đối với B được định nghĩa bởi*

$$SIG_B(b) = ES(D|B) - ES(D|B \cup \{b\}) \quad (1.19)$$

Theo Mệnh đề 3, ta có $ES(D|B) \geq ES(D|B \cup \{b\})$nên $SIG_B(b) \geq 0$. Do đó, $SIG_B(b)$ Khi lượng thay đổi entropy càng lớn, thuộc tính b trở nên càng quan trọng hơn và ngược lại. Độ quan trọng của thuộc tính b đặc trưng cho khả năng phân lớp của nó vào các lớp quyết định. Do đó, thuộc tính b thường được sử dụng làm tiêu chuẩn trong thuật toán heuristic để lựa chọn TRG trong BQĐ đầy đủ. Để mô tả thuật toán heuristic sử dụng entropy Shannon để tìm TRG, ta có thể áp dụng hai hướng tiếp cận: từ dưới lên (bottom-up) và từ trên xuống (top-down). Phần này sẽ mô tả một thuật toán heuristic tính toán lõi theo hướng tiếp cận từ dưới lên. Ý tưởng của thuật toán là bắt đầu từ tập lõi $HCOREC$, sau đó tiếp tục tăng cường các thuộc tính có tính quan trọng lớn nhất cho đến khi tìm được TRG. Trình tự của thuật toán được trình bày trong bảng mã giả 4 dưới đây:

Algorithm 4: Find the core set based on the entropy Shannon

Input: $DS = (U, C \cup D)$

Output: $HCORE(C)$

1. Initialize: $HCORE(C) = 0$
2. **for** $c \in C$ **do**
3. | compute $ES(D|C\backslash\{c\})$
4. | **if** $ES(D|C\backslash\{c\}) \neq ES(D|C)$, **then** $HCORE(C) := HCORE(C) \cup \{c\}$
5. **end for**
6. **return** $HCORE(C)$

Xét BQĐ $DS = (U, C \cup D)$, với $B \subset C$ và $b \in C\backslash B$, giả sử rằng $\frac{U}{B} = \{B_1, B_2, \dots, B_k\}$, $\frac{U}{B\cup\{b\}} = \{B'_1, B'_2, \dots, B'_z\}$. Theo Định nghĩa 12 được trình bày ở trên, entropy Shannon có điều kiện của D khi đã biết tập thuộc tính C là $ES(D|C) = -\sum_{i=1}^{m} \frac{|C_i|}{|U|} \sum_{j=1}^{n} \frac{|C_i \cap D_j|}{|C_i|} log_2 \frac{|C_i \cap D_j|}{|C_i|}$. Để tính phân hoạch $U\backslash B \cup \{b\}$ khi biết phân hoạch $\frac{U}{B}$ sử dụng Thuật toán được trình bày trong bảng mã giả 5 như sau:

Algorithm 5: Compute $\frac{U}{B\cup\{b\}}$ based on $\frac{U}{B}$

Input: $\frac{U}{B} = \{B_1, B_2, \dots, B_k\}$

Output: $\frac{U}{B\cup\{b\}}$

1. Initialize: $TMP = \phi$
2. **for** $B_i \in U$ **do**
3. | compute $B_i/\{b\}$
4. | $TMP := TMP \cup B_i/\{b\}$
5. **end for**
6. **return** *TMP*

Dựa vào hai thuật toán trên, thuật toán heuristic tìm TRG tốt nhất trên BQĐ sử dụng entropy Shannon có điều kiện có tính toán lõi được trình bày như sau.

Algorithm 6: Conditional Entropy Based Algorithm for Reduction of Knowledge with Computing Core

Input: $DS = (U, C \cup D), B \subset C, b \in C \backslash B$

Output: A reduct B

1. Find the core set $HCORE(C)$based on the Algorithm 4.

// Find the entropy Shannon reduct

2. $$B = HCORE(C)$$

// Supplements one attribute with the highest significance into B

3. **while** $ES(D|B) \neq ES(D|C)$ **do**
4. | **for** $b \in C \backslash B$ **do**
5. | | compute $ES(D|B \backslash \{b\})$
6. | | select b_0 which satisfies: $SIG_B(b) = \underset{b \in C \backslash B}{Max}\{SIG_B(b)\}$
7. | | $$B := B \cup \{b_0\}$$
8. | | Compute $ES(D|B)$
9. | **end for**
10. **end while**

// Remove the redundant attribute in B

11. $$B^* = B \backslash HCORE(C)$$
12. **for** $b \in B^*$ **do**
13. | compute $ES(D|B \backslash \{b\})$
14. | **if** $ES(D|B \backslash \{b\}) = ES(D|C)$ then $B := B \backslash \{b\}$
15. **end for**
16. **return** B

1.2.2. Phương pháp rút gọn thuộc tính theo tiếp cận tập mờ

Các nghiên cứu đã chỉ ra rằng phương pháp RGTT dựa trên tiếp cận tập thô là hiệu quả trên các BQĐ có thuộc tính giá trị rời rạc. Tuy nhiên, đối với các BQĐ có thuộc tính giá trị liên tục (BQĐ số), việc chuyển đổi miền giá trị từ liên tục sang rời rạc là cần thiết trước khi áp dụng RGTT. Quá trình này có thể tạo ra chi phí thực hiện và có thể dẫn đến mất mát dữ liệu. Vì vậy, các nhà nghiên cứu đã đề xuất phương pháp RGTT trực tiếp trên các BQĐ gốc mà không cần phải thực hiện bước rời rạc hóa dữ liệu trước. Một trong những phương pháp này là các phương pháp RGTT dựa trên tiếp cận TTM.

Các phương pháp dựa trên TTM tìm rút gọn trực tiếp trên dữ liệu gốc dựa trên QHTĐM. Vì QHTĐM bảo toàn sự khác biệt của các đối tượng, nên cách tiếp cận TTM có khả năng tăng cường độ chính xác khi phân loại rút. Trong những năm gần đây, RGTT dựa trên TTM đã thu hút nhiều tác giả. Một số phương pháp điển hình của phương pháp này là hàm phụ thuộc mờ [10], [11], [12, 13, 14, 15], miền dương mờ [16, 17, 18], ma trận mờ phân biệt [19, 20], entropy mờ [21, 22, 23, 24], khoảng cách mờ [25, 26, 27] và một số phương pháp khác, chẳng hạn như độ chi tiết của thông tin mờ [28], mức tăng thông tin mờ [29]. Trong phần này, báo cáo sẽ trình bày một số thuật toán trong việc tìm kiếm một rút gọn trên BQĐ đầy đủ chưa biến động và BQĐ khi có sự thay đổi số lượng đối tượng theo hướng tiếp cận tập mờ sử dụng độ đo khoảng cách mờ.

- Thuật toán tìm tập rút gọn dựa trên khoảng cách mờ trước khi gia tăng

Định nghĩa 16. [9] *Cho BQĐ* $DS = (U, C \cup D)$, *trong đó* $U = \{u_1, u_2, \dots, u_n\}$, $P, Q \subseteq C$ *và hai phân hoạch mờ trên* P *và* Q *là* $\tilde{\Upsilon}_P = \{[\widetilde{u_i}]_P\}$ *và* $\tilde{\Upsilon}_Q = \{[\widetilde{u_i}]_Q\}$*với* $u \in U$, *khi đó khoảng cách giữa hai phân hoạch* $\tilde{\Upsilon}_P$ *và* $\tilde{\Upsilon}_Q$ *là:*

$$\varphi(\tilde{\Upsilon}_P, \tilde{\Upsilon}_Q) = \frac{1}{|U|}\sum_{i=1}^{|U|}\left(\frac{|[\widetilde{u_i}]_P \cup [\widetilde{u_i}]_P| - |[\widetilde{u_i}]_P \cap [\widetilde{u_i}]_P|}{|U|}\right) \quad (1.20)$$

Mệnh đề 4. [9] *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$*và* $\tilde{R}$ *là một QHTĐM được định nghĩa bởi miền giá trị của thuộc tính điều kiện. Khoảng cách mờ giữa hai tập thuộc tính* C *và* $C \cup D$ *là:*

$$\varphi\left(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}\right) = \frac{1}{n^2}\sum_{i=1}^{n}\left(\left|[\widetilde{u_i}]_C\right| - \left|[\widetilde{u_i}]_C \cap |[\widetilde{u_i}]_D|\right|\right) \quad (1.21)$$

Định nghĩa 17. [9] *Cho BQĐ* $DS = (U, C \cup D)$*, trong đó* $B \subset C$ *và* $b \in C \backslash B$*. Độ quan trọng của thuộc tính b với B được định nghĩa như sau:*

$$SIG_B(b) = \varphi\left(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}\right) - \varphi\left(\tilde{\Upsilon}_{B\cup\{b\}}, \tilde{\Upsilon}_{B\cup\{b\}\cup D}\right) \quad (1.22)$$

Định nghĩa 18. [9] *Cho BQĐ* $DS = (U, C \cup D)$ *và* $\tilde{R}_B, \tilde{R}_C$ *là hai QHTĐM trên tập thuộc tính B và C với* $B \subset C$*. Khi đó B được gọi là một rút gọn của BQĐ sử dụng khoảng cách mờ nếu thỏa mãn:*

1. $\varphi\left(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}\right) = \varphi\left(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}\right)$
2. $\forall b \in B, \varphi\left(\tilde{\Upsilon}_{B\backslash\{b\}}, \tilde{\Upsilon}_{B\backslash\{b\}\cup D}\right) = \varphi\left(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}\right)$

Từ một số định nghĩa và mệnh đề trên, [9] đã thiết kế thuật toán Fuzzy Distance Attribute Reduction (FDAR) nhằm tìm kiếm một rút gọn trên BQĐ ban đầu.

Algorithm 7: Fuzzy Distance Attribute Reduction (FDAR)

Input: $DS = (U, C \cup D)$ and $\tilde{R}$

Output: A reduct B

1. **while** $\varphi\left(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}\right) \neq \varphi\left(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}\right)$ **do**
2. | **for** $b \in C \backslash B$ **do**
3. | | select b_0 which satisfies: $SIG_B(b) = \underset{b\in C\backslash B}{Max}\{SIG_B(b)\}$
4. | | $B := B \cup \{b_0\}$
5. | **end for**
6. **end while**
7. **return** B

- Thuật toán gia tăng tìm tập rút gọn dựa trên khoảng cách mờ

Mệnh đề 5. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$ *và* $\tilde{R}$ *là một QHTĐM được định nghĩa trên miền giá trị tập thuộc tính điều kiện. Giả sử rằng, TĐT mới bao gồm s phần tử* $\Delta U = \{u_{n+1}, u_{n+2}, \dots, u_{n+s}\}$ *được thêm vào U. Với* $M_{U\cup\Delta U}(\tilde{R}_C) = [m_{ij}]_{(n+s)(n+s)}$, $M_{U\cup\Delta U}(\tilde{R}_D) = [d_{ij}]_{(n+s)(n+s)}$ *là hai ma trận tương đương trên C và D, công thức gia tăng tính khoảng cách được trình bày như sau:*

$$\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \left(\frac{n}{n+s}\right)^2 \varphi_U(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) + \frac{2}{(n+s)^2}\sum_{i=1}^{s}\left(\left|\widetilde{[u_{n+i}]}_C\right| - \left|\widetilde{[u_{n+i}]}_C \cap \widetilde{[u_{n+i}]}_D\right| - \alpha_i\right) \quad (1.23)$$

trong đó, $a_i = \sum_{j=1}^{s-1}\left(m_{n+i,n+j+1} - min\left(m_{n+i,n+j+1}, d_{n+i,n+j+1}\right)\right)$.

Mệnh đề 6. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$ *và* $\tilde{R}$ *là một QHTĐM,* $B \subseteq C$ *là một rút gọn dựa trên khoảng cách mờ. Giả sử rằng TĐT* $\Delta U = \{u_{n+1}, u_{n+2}, \dots, u_{n+s}\}$ *được bổ sung vào U. Khi đó, chúng ta có hai trường hợp sau:*

1. *Nếu* $D(u_{n+1}) = d$ *với* $i = 1,2, \dots, s$ *thì*

$$\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \left(\frac{n}{n+s}\right)^2 \varphi_U(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) + \frac{2}{(n+s)^2}\sum_{i=1}^{s}\left(\left|\widetilde{[u_{n+i}]}_C\right| - \left|\widetilde{[u_{n+i}]}_C \cap \widetilde{[u_{n+i}]}_D\right|\right) \quad (1.24)$$

2. *Nếu* $\widetilde{[u_{n+i}]}_B \subseteq \widetilde{[u_{n+i}]}_D$ với $i = 1,2, \dots, s$ *thì*

$$\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \varphi_{U\cup\Delta U}(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}) \quad (1.25)$$

Dựa trên các định nghĩa và mệnh đề nêu trên, thuật toán gia tăng tìm TRG trên BQĐ trong trường hợp bổ sung TĐT được trình bày như sau:

Algorithm 8: Incremental Filter Algorithm for Finding Reduct After Adding a Set of Objects (IF-FDAR-AdObjs)

Input:

$DS = (U, C \cup D)$, a reduct $B \subseteq C$ and $\tilde{R}$

$M_u(\tilde{R}_B) = [b_{ij}]_{n\times n}$, $M_u(\tilde{R}_C) = [c_{ij}]_{n\times n}$, $M_u(\tilde{R}_D) = [d_{ij}]_{n\times n}$

$$\Delta U = \{u_{n+1}, u_{n+2}, \dots, u_{n+s}\}$$

Output: The approximation reduct B of $DS' = (U \cup \Delta U, C \cup D)$

// Initialization

1. $B := \phi$
2. compute fuzzy equivalence matrices on the object set $U \cup \Delta U$ $M_{U\cup\Delta U}(\tilde{R}_B) = [b_{ij}]_{(n+s)\times(n+s)}$, $M_{U\cup\Delta U}(\tilde{R}_D) = [d_{ij}]_{(n+s)\times(n+s)}$

// Check the added set of objects

3. $X := \Delta U$
4. **for** $i = 1$ to s **do**
5. | **if** $[\widetilde{u_{n+i}}]_B \subseteq [\widetilde{u_{n+i}}]_D$ **then** $X := X \backslash \{u_{n+i}\}$
6. | **if** $X = \phi$ **then** return B_0 ***//Approximation reduct does not change***
7. **end for**
8. set $\Delta U := X$, $S := \Delta U$ ***// reset the object set***

//Finding the reduct

9. compute $\varphi_U(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D})$, $\varphi_U(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D})$
10. compute $\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D})$, $\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D})$

//Filter stage

11. **while** $\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}) \neq \varphi_{U\cup\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D})$ **do**
12. | **for** $b \in C \backslash B$ **do**
13. | | compute $\varphi_{U\cup\Delta U}(\tilde{\Upsilon}_{B\cup\{b\}}, \tilde{\Upsilon}_{B\cup\{b\}\cup D})$by incremental forrmulas
14. | select b_0 which satisfies: $SIG_B(b_0) = \underset{b\in C\backslash B}{Max}\{SIG_B(b)\}$
15. | $B := B \cup \{b_0\}$
16. | **end for**
17. **end while**

18. **return** B

Mệnh đề 7. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$ *và* $\tilde{R}$ *là một QHTĐM được định nghĩa trên miền giá trị tập thuộc tính điều kiện. Giả sử rằng, TĐT* $\Delta U = \{u_k, u_{k+1}, \dots, u_{k+s-1}\}$ *bị loại bỏ khỏi U. Các ma trận tương đương mờ trên C và D của BQĐ khi bị loại bỏ lần lượt là* $M_{U\setminus\Delta U}(\tilde{R}_C) = [m_{ij}]_{(n-s)(n-s)}$ *và* $M_{U\setminus\Delta U}(\tilde{R}_D) = [d_{ij}]_{(n-s)(n-s)}$. *Công thức gia tăng khoảng cách được trình bày như sau:*

$$\varphi_{U\setminus\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \left(\frac{n}{n-s}\right)^2 \varphi_U(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) - \frac{2}{(n-s)^2}\sum_{i=0}^{s-1}\left(\left|\widetilde{[u_{n+i}]}_C\right| - \left|\widetilde{[u_{n+i}]}_C \cap \widetilde{[u_{n+i}]}_D\right| - \beta_i\right) \quad (1.26)$$

trong đó, $\beta_i = \sum_{j=0}^{i}(m_{k+i,k+j} - min(m_{k+i,k+j}, d_{k+i,k+j}))$.

Mệnh đề 8. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \dots, u_n\}$ *và* $\tilde{R}$ *là một QHTĐM,* $B \subseteq C$ *là một rút gọn dựa trên khoảng cách mờ. Giả sử rằng TĐT* $\Delta U = \{u_k, u_{k+1}, \dots, u_{k+s-1}\}$ *được loại bỏ khỏi U. Khi đó, chúng ta có hai trường hợp sau:*

1. Nếu $D(u_{k+i}) = d$ *với* $i = 1,2, \dots, s-1$ *thì*

$$\varphi_{U\setminus\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \left(\frac{n}{n-s}\right)^2 \varphi_U(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) - \frac{2}{(n-s)^2}\sum_{i=0}^{s-1}\left(\left|\widetilde{[u_{n+i}]}_C\right| - \left|\widetilde{[u_{n+i}]}_C \cap \widetilde{[u_{n+i}]}_D\right|\right) \quad (1.27)$$

2. Nếu $\widetilde{[u_{k+i}]}_B \subseteq \widetilde{[u_{k+i}]}_D$ với *i=1,2,...,s-1 thì*

$$\varphi_{U\setminus\Delta U}(\tilde{\Upsilon}_C, \tilde{\Upsilon}_{C\cup D}) = \varphi_{U\setminus\Delta U}(\tilde{\Upsilon}_B, \tilde{\Upsilon}_{B\cup D}) \quad (1.28)$$

Cũng tương tự như thuật toán IF-FDAR-AdObjs, thuật toán gia tăng tìm rút gọn trên BQĐ khi loại bỏ TĐT được trình bày trong bảng mã giả số 9.

Các phương pháp RGTT trực tiếp trên BQĐ số hiện nay đa phần chỉ dựa trên tiếp cận TTM. Các kết quả thực nghiệm đã cho thấy TRG thu được theo tiếp cận này

còn chưa hiệu quả về kích thước và độ chính xác phân lớp trên các bộ dữ liệu nhiễu do không gian xấp xỉ mờ là chưa đủ để mô tả mối quan hệ của các đối tượng trong một tập. Đối với phương pháp RGTT theo tiếp cận tập mờ, trên thế giới hiện nay chưa được biết tới mặc dù cách thức xây dựng không gian xấp xỉ mờ phản ánh đầy đủ thông tin quan hệ của một đối tượng và độ đo đánh giá độ quan trọng của thuộc tính mang tính chặt chẽ. Trong phần sau của báo cáo này, luận văn sẽ nêu rõ về lý thuyết tập mờ và đề xuất hướng xây dựng một số thuật toán RGTT theo cách tiếp cận tập mờ.

CHƯƠNG 2. LÝ THUYẾT TẬP MỜ MỨC α VÀ MỘT SỐ THUẬT TOÁN GIA TĂNG RÚT GỌN THUỘC TÍNH

2.1. MỘT SỐ KHÁI NIỆM CƠ BẢN

Như đã trình bày ở các phần trên, lý thuyết tập thô không hiệu quả khi xử lý với các bảng dữ liệu mang miền giá trị số, liên tục. Lý thuyết tập mờ không hiệu quả khi xử lý với các BQĐ có độ chính xác ban đầu thấp do sự hạn chế về khả năng loại bỏ nhiễu. Do đó, để giải quyết vấn đề này, đầu tiên đề tài sẽ xây dựng một tập lát cắt α làm cơ sở để xây dựng các lớp tương đương mờ mức α trong các phân hoạch của từng thuộc tính trên BQĐ. Sau đó, luận văn sẽ xây dựng hai công thức tính toán gia tăng nhằm tạo tiền đề cho việc xây dựng độ đo quan trọng của các thuộc tính. Cuối cùng, luận văn sẽ đề xuất hai thuật toán gia tăng để tìm kiếm các rút gọn trong trường hợp BQĐ có sự gia tăng hoặc loại bỏ TĐT.

Đầu tiên, xét BQĐ $DS = (U, C \cup D)$ với $U = \{u_1, u_2, \dots, u_n\}$, $A \subseteq C$ và $\widetilde{R_A}$ là một QHTĐM được định nghĩa trên miền giá trị của tập thuộc tính A. Cho α là một số thực nằm trong khoảng [0,1]. Khi đó, tập lát cắt α là một tập nguyên thủy dựa trên mức α của tập mờ $[\tilde{u}]_A$, ký hiệu là $[u]_A^\alpha$, được xác định như sau:

$$[u]_A^\alpha = \{v \in U : [\tilde{u}]_A(v) \geq \alpha\}$$

Tiếp theo, tập $[\tilde{u}]_A^\alpha$ được xây dựng bằng cách tổng hợp hợp các phần tử của $[u]_A^\alpha$ thông qua độ tương tự. Cụ thể, $[\tilde{u}]_A^\alpha$ là một tập mờ trên U với mỗi mức tương tự của mỗi đối tượng $v \in U$.

$$[\tilde{u}]_A^\alpha(v) = \begin{cases} [\tilde{u}]_A(v) \; v \in [u]_A^\alpha \\ 0 \;\; \textit{với các trường hợp còn lại} \end{cases}$$

Dễ thấy rằng, $[\tilde{u}]_A^\alpha$ sẽ được hình thành dựa trên việc điều chỉnh các số mờ từ lớp tương đương mờ $[u]_A^\alpha$. Những số mờ này có mức tương tự nhỏ hơn α. Trong luận văn này, chúng tôi sẽ gọi $[\tilde{u}]_A^\alpha$ là một lớp tương đương mờ mức α của đối tượng u. Do đó, một họ $\{[\tilde{u}]_A^\alpha : u \in U\}$ sẽ tạo ra một phân hoạch mờ trên U. Một cách đơn giản, họ này sẽ được ký hiệu là $\widetilde{\Upsilon_A^\alpha}$ và được gọi là phân hoạch mờ mức α.

Cho $\widetilde{\Upsilon}_A^\alpha$ và $\widetilde{\Upsilon}_B^\alpha$ là hai phân hoạch mờ mức α trên tập thuộc tính A và B. Chúng tôi nói rằng $\widetilde{\Upsilon}_A^\alpha$ mịn hơn $\widetilde{\Upsilon}_B^\alpha$, ký hiệu là $\widetilde{\Upsilon}_A^\alpha \preccurlyeq \widetilde{\Upsilon}_B^\alpha$ nếu với mọi đối tượng $u \in U$, $[\tilde{u}]_A^\alpha \subseteq [\tilde{u}]_B^\alpha$. Tiếp theo, luận văn sẽ trình bày một số tính chất của phân hoạch mờ và lớp tương đương mức α.

Mệnh đề 1. Cho BQĐ $DS = (U, C \cup D)$

(i) Nếu $A, B \subseteq C$ *thì* $[\tilde{u}]_{A\cup B}^\alpha = [\tilde{u}]_A^\alpha \cap [\tilde{u}]_B^\alpha$.

(ii) Nếu $A \subseteq B$ *thì* $\widetilde{\Upsilon}_A^\alpha \preccurlyeq \widetilde{\Upsilon}_B^\alpha$.

(iii) Nếu $\alpha_1 \leq \alpha_2$ *thì* $\widetilde{\Upsilon}_A^{\alpha_2} \preccurlyeq \widetilde{\Upsilon}_A^{\alpha_1}$

2.2. THUẬT TOÁN RÚT GỌN THUỘC TÍNH TRÊN BẢNG QUYẾT ĐỊNH CỐ ĐỊNH

Định nghĩa 1. Cho BQĐ $DS = (U, C \cup D)$ với TĐT $U = \{u_1, u_2, \dots, u_n\}$ và hai phân hoạch mờ mức α là $\widetilde{\Upsilon}_A^\alpha$ và $\widetilde{\Upsilon}_B^\alpha$ được hình thành bởi các lớp tương đương mờ mức α là $[\tilde{u}]_A^\alpha$ và $[\tilde{u}]_B^\alpha$ của tập thuộc tính $A, B \subseteq C$. Với mọi $u_i \in U$, khoảng cách phân hoạch mờ giữa $\widetilde{\Upsilon}_A^\alpha$ và $\widetilde{\Upsilon}_B^\alpha$ ký hiệu là $\widetilde{D}(\widetilde{\Upsilon}_A^\alpha, \widetilde{\Upsilon}_B^\alpha)$ được xác định như sau:

$$\widetilde{D}(\widetilde{\Upsilon}_A^\alpha, \widetilde{\Upsilon}_B^\alpha) = \sum_{i=1}^{n} \frac{|[\tilde{u}_i]_A^\alpha \cup [\tilde{u}_i]_A^\alpha| - |[\tilde{u}_i]_A^\alpha \cap [\tilde{u}_i]_A^\alpha|}{n^2}$$

Mệnh đề 2. Cho BQĐ $DS = (U, C \cup D)$ với TĐT $U = \{u_1, u_2, \dots, u_n\}$. Với mọi $u_i \in U$, khoảng cách phân hoạch mờ giữa hai phân hoạch mờ mức α được tạo bởi tập thuộc tính C và $C \cup D$ được xác định như sau:

$$\widetilde{D}(\widetilde{\Upsilon}_C^\alpha, \widetilde{\Upsilon}_{C\cup D}^\alpha) = \sum_{i=1}^{n} \frac{|[\tilde{u}_i]_C^\alpha| - |[\tilde{u}_i]_C^\alpha \cap [\tilde{u}_i]_D^\alpha|}{n^2}$$

Mệnh đề 3. Cho BQĐ $DS = (U, C \cup D)$ và $A, B \subseteq C$. Nếu $A \subseteq B$ thì $\widetilde{D}(\widetilde{\Upsilon}_A^\alpha, \widetilde{\Upsilon}_{A\cup D}^\alpha) \geq \widetilde{D}(\widetilde{\Upsilon}_B^\alpha, \widetilde{\Upsilon}_{B\cup D}^\alpha)$.

Định nghĩa 2. Cho BQĐ $DS = (U, C \cup D)$, khi đó một tập con B được gọi là một rút gọn của C nếu thỏa mãn:

(i) $\widetilde{D}(\widetilde{\Upsilon}_B^\alpha, \widetilde{\Upsilon}_{B\cup D}^\alpha) = \widetilde{D}(\widetilde{\Upsilon}_C^\alpha, \widetilde{\Upsilon}_{C\cup D}^\alpha)$

(ii) $\forall B' \subset B, \widetilde{D}(\widetilde{\Upsilon}_{B'}^\alpha, \widetilde{\Upsilon}_{B'\cup D}^\alpha) > \widetilde{D}(\widetilde{\Upsilon}_B^\alpha, \widetilde{\Upsilon}_{B\cup D}^\alpha)$.

Định nghĩa 3. Cho BQĐ $DS = (U, C \cup D)$, một tập con thuộc tính B và một thuộc tính $b \in C \backslash B$, khi đó độ quan trọng của thuộc tính b theo B được xác định như sau:

$$Sig_B(b) = \widetilde{D}\left(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}}\right) - \widetilde{D}\left(\widetilde{\Upsilon_{B\cup\{b\}}^{\alpha}}, \widetilde{\Upsilon_{B\cup\{b\}\cup D}^{\alpha}}\right)$$

Theo tính chất của khoảng cách mờ (Mệnh đề 3) ta có $Sig_B(b)$. Độ quan trọng $Sig_B(b)$ đặc trưng cho chất lượng phân lớp của thuộc tính b đối với thuộc tính quyết định D và được sử dụng làm tiêu chuẩn lựa chọn thuộc tính cho thuật toán filter F_FDBAR_α tìm TRG.

Thuật toán F_FDBAR_α (Filter - Fuzzy Distance Based Attribute Reduction α): Thuật toán filter tìm tập rút gọn sử dụng khoảng cách mờ.

Đầu vào: Bảng quyết định $DS = (U, C \cup D)$, QHTĐM $\tilde{R}$ xác định trên tập thuộc tính điều kiện.

Đầu ra: Một tập rút gọn B

1. $B \leftarrow \emptyset$; $\widetilde{D}\left(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}}\right) = 1$;
2. Tính khoảng cách mờ $\widetilde{D}\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right)$;

// Thêm dần vào B các thuộc tính có độ quan trọng lớn nhất

3. While $\widetilde{D}\left(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}}\right) \neq \widetilde{D}\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right)$ do
4. Begin
5. Với mỗi $a \in C - B$ tính

   $$Sig_B(a) = \widetilde{D}\left(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}}\right) - \widetilde{D}\left(\widetilde{\Upsilon_{B\cup\{a\}}^{\alpha}}, \widetilde{\Upsilon_{B\cup\{a\}\cup D}^{\alpha}}\right)$$

6. Chọn $a_m \in C - B$ sao cho $SIG_B(a_m) = \underset{a\in C-B}{Max}\{SIG_B(a)\}$;
7. $B = B \cup \{a_m\}$;
8. End;

Tiếp theo, luận văn đánh giá độ phức tạp thời gian của thuật toán F_FDBAR_α, gọi tắt là độ phức tạp. Giả sử $D = \{d\}$ và ký hiệu $|C|, |U|$ tương ứng là số thuộc tính

điều kiện và số đối tượng. Độ phức tạp tính ma trận tương đương mờ $M(\widetilde{\Upsilon_C^{\alpha}})$ là $O(|C||U|^2)$, do đó độ phức tạp tính khoảng cách mờ trong câu lệnh 2 là $O(|C||U|^2)$. Xét vòng lặp While từ câu lệnh 3 đến 8, để tính $SIG_B(a)$ ta phải tính $\widetilde{D}(\widetilde{\Upsilon_{B\cup\{a\}}^{\alpha}}, \widetilde{\Upsilon_{B\cup\{a\}\cup D}^{\alpha}})$ vì $\widetilde{D}(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}})$ đã được tính ở bước trước. Độ phức tạp tính$\widetilde{D}(\widetilde{\Upsilon_{B\cup\{a\}}^{\alpha}}, \widetilde{\Upsilon_{B\cup\{a\}\cup D}^{\alpha}})$ bằng độ phức tạp tính ma trận tương đương mờ của thuộc tính a, nghĩa là $O(|U|^2)$. Do có hai vòng lặp lồng nhau theo $|C|$ nên độ phức tạp của vòng lặp While là $O(|C|^2|U|^2)$. Tương tự, độ phức tạp của vòng lặp For từ dòng lệnh số 9 đến 13 là $O(|C|^2|U|^2)$. Do đó, độ phức tạp của thuật toán F_FDBAR_α là $O(|C|^2|U|^2)$

Xét BQĐ $DS = (U, C \cup D)$ với $C = \{a_1, a_2, \ldots, a_m\}$ và $\tilde{R}$ là QHTĐM xác định trên miền giá trị thuộc tính điều kiện. Đặt $\omega = \widetilde{D}(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}})$. Theo thuật toán F_FDBAR_α, giả sử các thuộc tính $a_{i_1}, a_{i_2}, \ldots$ được thêm vào tập rỗng theo giá trị lớn nhất của độ quan trọng thuộc tính cho đến khi tồn tại $t \in \{1,2,\ldots m\}$ sao cho $\widetilde{D}\left(\left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2},\ldots,a_{i_t}\}}\right), \left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2},\ldots,a_{i_t}\}\cup D}\right)\right) = \omega$. Kết thúc thuật toán, ta thu được TRG $B = \{a_{i_1}, a_{i_2}, \ldots, a_{i_t}\}$, độ chính xác phân lớp trên tập dữ liệu được tính bởi độ chính xác phân lớp trên B. Do đó, thuật toán F_FDBAR_α theo hướng tiếp cận filter truyền thống.

Mặt khác, theo Mệnh đề 3 ta có $\widetilde{D}\left(\tilde{Y}^{\alpha}_{\{a_{i_1}\}}, \tilde{Y}^{\alpha}_{\{a_{i_1}\}\cup D}\right) \geq \widetilde{D}\left(\left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2}\}}\right), \left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2}\}\cup D}\right)\right) \geq \ldots \geq \widetilde{D}\left(\left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2},\ldots,a_{i_t}\}}\right), \pi\left(\tilde{Y}^{\alpha}_{\{a_{i_1},a_{i_2},\ldots,a_{i_t}\}\cup D}\right)\right) = \omega$ Với ngưỡng $\varepsilon > \omega$ cho trước, đặt $B_k = \{a_{i_1}, \ldots, a_{i_k}\}$ thỏa mãn $\widetilde{D}\left(\left(\tilde{Y}^{\alpha}_{B_k}\right), \left(\tilde{Y}^{\alpha}_{B_k\cup D}\right)\right) \geq \varepsilon$ và $\widetilde{D}\left(\left(\tilde{Y}^{\alpha}_{B_k\cup\{a_{i_{k+1}}\}}\right), \left(\tilde{Y}^{\alpha}_{B_k\cup\{a_{i_{k+1}}\}}\right)\right) < \varepsilon$. Khi đó, B_k *được gọi là TRG xấp xỉ ngưỡng* ε. Nếu B_k và $B_k \cup \{a_{i_{k+1}}, \ldots, a_{i_t}\}$ được sử dụng để xây dựng bộ phân lớp, công bố [9] cho thấy, độ chính xác phân lớp trên $B_k \cup$

$\{a_{i_{k+1}}, \ldots, a_{i_t}\}$ chưa chắc đã tốt hơn trên B_k. Giả sử B_k có độ chính xác phân lớp tốt hơn $B_k \cup \{a_{i_{k+1}}, \ldots, a_{i_t}\}$. Khi đó, nếu chọn B_k là kết quả của thuật toán thì B_k có độ chính xác phân lớp cao hơn, có số lượng thuộc tính ít hơn nên khả năng khái quát hóa và hiệu năng thực hiện các thuật toán phân lớp sẽ cao hơn.

2.3. THUẬT TOÁN GIA TĂNG FIFTER TÌM TẬP RÚT GỌN KHI BỔ SUNG TẬP ĐỐI TƯỢNG

Công thức gia tăng tính khoảng cách mờ khi bổ sung tập đối tượng

Từ Mệnh đề 4, luận văn giới thiệu công thức gia tăng tính khoảng cách mờ khi thêm một TĐT ở Mệnh đề 5:

Mệnh đề 5. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{x_1, x_2, \ldots, x_n\}$ *và* $\tilde{Y}$ *là QHTĐM xác định trên miền giá trị tập thuộc tính điều kiện. Giả sử TĐT gồm s phần tử* $\Delta U = \{x_{n+1}, x_{n+2}, \ldots, x_{n+s}\}$ *được bổ sung vào U, mà s≥2. Với* $M_{U\cup\Delta U}(\widetilde{\Upsilon_C^{\alpha}}) = [m_{ij}]_{(n+s)\times(n+s)}$, $M_{U\cup\Delta U}(\widetilde{\Upsilon_D^{\alpha}}) = [d_{ij}]_{(n+s)\times(n+s)}$ *là ma trận tương đương mờ tương ứng trên C và D. Khi đó, công thức gia tăng khoảng cách mờ như sau:*

$$\widetilde{D}_{U\cup\Delta U}(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}) = \left(\frac{n}{n+s}\right)^2 \widetilde{D}_U(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}) + \frac{2}{(n+s)^2}\sum_{i-1}^{s}((|[x_{n+i}]_{\tilde{C}}| - |[x_{n+i}]_{\tilde{C}} \cap [x_{n+i}]_{\tilde{D}}|) - \alpha_i)$$

mà $\alpha_i = \sum_{j=i}^{s-1}\left(m_{n+i,n+j+1} - \min\left(m_{n+i,n+j+1}, d_{n+i,n+j+1}\right)\right)$

Thuật toán gia tăng fifter tìm tập rút gọn sau khi bổ sung tập đối tượng:

Mệnh đề 6. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{x_1, x_2, \ldots, x_n\}$ *và* $\tilde{R}$ *là QHTĐM xác định trên miền giá trị tập thuộc tính điều kiện,* $B \subseteq C$ *là TRG dựa trên khoảng cách mờ. Giả sử TĐT gồm s phần tử* $\Delta U = \{x_{n+1}, x_{n+2}, \ldots, x_{n+s}\}$ *được bổ sung vào U. Khi đó ta có:*

1) Nếu $D(x_{n+i}) = d$ *với mọi* $i = 1,2, \ldots, s$ *thì:*

2) $\widetilde{D}_{U\cup\Delta U}(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}) = \left(\frac{n}{n+s}\right)^2 \widetilde{D}_U(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}) + \frac{2}{(n+s)^2}\sum_{i-1}^{s}((|[x_{n+i}]_{\tilde{C}}| - |[x_{n+i}]_{\tilde{C}} \cap [x_{n+i}]_{\tilde{D}}|))$

3) *Nếu* $[x_{n+i}]_{\tilde{B}} \subseteq [x_{n+i}]_{\tilde{D}}$ *với mọi* $i = 1,2,\ldots,s$ *thì* $\widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}})\right) = \widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}})\right)$.

Từ kết quả của Mệnh đề 6, thuật toán gia tăng filter-wrapper RGTT sử dụng khoảng cách mờ IF_FDAR_AdObj_α gồm 3 bước chính:

Algorithm IF_FDAR_AdObj_α

Đầu vào:

1. Bảng quyết định $DS = (U, C \cup D)$ với $U = \{x_1, x_2, \ldots, x_n\}$, QHTĐM $\tilde{R}$, tập rút gọn $B \subseteq C$.
2. Các ma trận tương đương mờ

$$M_U(\tilde{R}_B) = [b_{ij}]_{n\times n}, M_U(\widetilde{\Upsilon_C^{\alpha}}) = [c_{ij}]_{n\times n}, M_U(\widetilde{\Upsilon_D^{\alpha}}) = [d_{ij}]_{n\times n}$$

3. Tập đối tượng bổ sung $\Delta U = \{x_{n+1}, x_{n+2}, \ldots, x_{n+s}\}$

Đầu ra: Tập rút gọn xấp xỉ B_{best} của $DS' = (U \cup \Delta U, C \cup D)$ với độ chính xác phân loại cao nhất.

Bước 1: Khởi tạo

1. $T := \emptyset$; *// T chứa ứng của viên tập rút gọn tốt nhất*
2. Tính các ma trận tương đương mờ trên tập đối tượng $U \cup \Delta U$

$$M_{U\cup\Delta U}(\tilde{R}_B) = [b_{ij}]_{(n+s)\times(n+s)}, M_{U\cup\Delta U}(\widetilde{\Upsilon_D^{\alpha}}) = [d_{ij}]_{(n+s)\times(n+s)};$$

Bước 2: Kiểm tra tập đối tượng thêm vào

3. Đặt $X := \Delta U$;
4. For $i = 1$ to s do
5. If $[x_{n+i}]_{\tilde{B}} \subseteq [x_{n+i}]_{\tilde{D}}$ then $X := X - \{x_{n+i}\}$;
6. If $X = \emptyset$ then Return B_0; *// Tập xấp xỉ không thay đổi*
7. Đặt $\Delta U := X$; $s := |\Delta U|$; *//Gán lại tập đối tượng*

Bước 3: Tìm tập rút gọn tốt nhất

8. Tính các khoảng cách mờ ban đầu

$\widetilde{D}_U(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha})$; $\widetilde{D}_U\left((\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha})\right)$;

9. Tính khoảng cách mờ bởi công thức gia tăng:

$$\widetilde{D}_{U\cup\Delta U}(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha});\ \widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha})\right)$$

// Giai đoạn fifter: tìm các ứng viên cho tập rút gọn

10. While $\widetilde{D}_{U\cup\Delta U}(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha}) \neq \widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha})\right)$ do
11. Begin
12. For each $a \in C - B$ do
13. Begin
14. Tính $\widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_{B\cup\{a\}}^\alpha}), (\widetilde{\Upsilon_{B\cup\{a\}\cup D}^\alpha})\right)$ bởi công thức gia tăng;
15. Tính $SIG_B(a) = \widetilde{D}_{U\cup\Delta U}(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha}) - \widetilde{D}_{U\cup\Delta U}\left((\widetilde{\Upsilon_{B\cup\{a\}}^\alpha}), (\widetilde{\Upsilon_{B\cup\{a\}\cup D}^\alpha})\right)$;
16. End;
17. Select$a \in C - B$ satisfying $SIG_B(a_m) = \underset{a\in C-B}{Max}\{SIG_B(a)\}$;
18. $B := B \cup \{a_m\}$;
19. $B_0 := B_0 \cup \{a_m\}$;
20. $T := T \cup B_0$;
21. End;

Trong phần này, Luận văn sẽ đánh giá độ phức tạp của thuật toán IF_FDAR_AdObj_α . Giả sử $D = \{d\}, |C|, |U|, |\Delta U|$ tương ứng là số thuộc tính điều kiện, số đối tượng và số đối tượng bổ sung từ tập ban đầu. Độ phức tạp của thuật toán được tính dựa trên thuật toán trên.

Độ phức tạp của ma trận tương đương mờ ở câu lệnh 2 trên$|U| + |\Delta U|$ là $O(|B| * |\Delta U| * (|U| + |\Delta U|))$và độ phức tạp của vòng for ở câu lệnh 4, 5 là

$O(|\Delta U| * (|U| + |\Delta U|))$. Trong trường hợp tốt nhất, thuật toán kết thúc ở câu lệnh 6 (TRG không thay đổi). Khi đó, độ phức tạp của thuật toán IF_FDAR_AdObj_α là$O(|B| * |\Delta U| * (|U| + |\Delta U|))$. Ngược lại, độ phức tạp của khoảng cách mờ ở câu lệnh 9 là$O(|C| * |\Delta U| * (|U| + |\Delta U|))$, độ phức tạp tính gia tăng $\widetilde{D}_{U+\Delta U}\left(\left(\widetilde{\Upsilon^{\alpha}_{B\cup\{a\}}}\right), \left(\widetilde{\Upsilon^{\alpha}_{B\cup\{a\}\cup\{d\}}}\right)\right)$là $O(|\Delta U| * (|U| + |\Delta U|))$. Bằng cách tính độ phức tạp tương tự như thuật toán FW_FDBAR ở trong phần 2.4, độ phức tạp của vòng lặp While (từ câu lệnh 10 đến câu lệnh 21) là $O((|C| - |B|)^2|\Delta U| * (|U| + |\Delta U|))$. Kết quả độ phức tạp của giai đoạn fifter trong trường hợp xấu nhất là$O((|C| - |B|)^2|\Delta U| * (|U| + |\Delta U|))$.

Nếu thực hiện thuật toán không gia tăng FW_FDBAR_α trực tiếp trên BQĐ có số đối tượng $U \cup \Delta U$, theo mục 2, độ phức tạp của FW_FDBAR_α là $O(|C|^2 * (|U| + |\Delta U|)^2) + O(|C| * T)$. Dựa trên kết quả này chúng ta thấy rằng thuật toán IF_FDAR_AdObj_α giảm thiểu đáng kể thời gian thực hiện, đặc biệt trong trường hợp TĐT $|U|$ lớn hoặc tập điều kiện $|C|$ lớn và $|B|$ nhỏ.

2.4. THUẬT TOÁN GIA TĂNG FIFTER TÌM TẬP RÚT GỌN KHI LOẠI BỎ TẬP ĐỐI TƯỢNG

Cập nhật khoảng cách mờ khi loại bỏ tập đối tượng

Trên cơ sở Mệnh đề 6, chúng tôi xây dựng công thức cập nhật khoảng cách mờ trong trường hợp loại bỏ TĐT bởi Mệnh đề 3.5 như sau:

Mệnh đề 7. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{x_1, x_2, \ldots, x_n\}$ *và* $\tilde{R}$ *là một QHTĐM. Giả sử TĐT gồm s phần tử* $\Delta U = \{x_k, x_{k+1}, \ldots, x_{k+s-1}\}$ *bị loại khỏi U,* $s < n$ *. Ma trận tương đương mờ và ma trận tương đương trên C và D tương ứng được xác định bởi* $M_{U-\Delta U}(\tilde{Y}_C) = [m_{ij}]_{(n-s)\times(n-s)}$, $M_{U-\Delta U}(R_D) = [d_{ij}]_{(n-s)\times(n-s)}$ *. Khi đó, công thức cập nhật khoảng cách mờ như sau:*

$$\widetilde{D}_{U\cup\Delta U}\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right)$$

$$= \left(\frac{n}{n-s}\right)^2 \widetilde{D}_U\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right)$$

$$- \frac{2}{(n-s)^2}\sum_{i=0}^{s-1}\left(\left(\left|[x_{n+i}]_{\tilde{C}}\right| - \left|[x_{n+i}]_{\tilde{C}} \cap [x_{n+i}]_{\tilde{D}}\right|\right) - \varepsilon_i\right)$$

Với $\varepsilon_i = \sum_{j=0}^{i}\left(m_{k+i,k+j} - \min\left\{m_{k+i,k+j}, d_{k+i,k+j}\right\}\right)$

Chứng minh: Ký hiệu $\widetilde{D}_1, \widetilde{D}_2, \ldots, \widetilde{D}_s$ tương ứng là khoảng cách mờ khi loại bỏ lần lượt các đối tượng$x_k, x_{k+1}, \ldots, x_{k+s-1}$ khỏi U và $\widetilde{D}_0$là khoảng cách mờ trên TĐT ban đầu U. Áp dụng Mệnh đề 3.4, ta có:

$$\widetilde{D}_1 = \left(\frac{n}{n-1}\right)^2 \widetilde{D}_0$$

$$+ \frac{2}{(n-1)^2}\left(\left(\left|[x_k]_{\tilde{C}}\right| - \left|[x_k]_{\tilde{C}} \cap [x_k]_{\tilde{D}}\right|\right)\right.$$

$$\left. - \left(m_{k,k} - min\left\{m_{k,k}, d_{k,k}\right\}\right)\right)$$

$$\widetilde{D}_2 = \left(\frac{n-1}{n-2}\right)^2 \widetilde{D}_1 +$$

$$\frac{2}{(n-2)^2}\left(\left(\left|[x_{k+1}]_{\tilde{C}}\right| - \left|[x_{k+1}]_{\tilde{C}} \cap [x_{k+1}]_{\tilde{D}}\right|\right) - \left(m_{k+1,k} - min\left\{m_{k+1,k}, d_{k+1,k}\right\}\right)\right.$$

$$\left. - \left(m_{k+1,k+1} - min\left\{m_{k+1,k+1}, d_{k+1,k+1}\right\}\right)\right)\widetilde{D}_2$$

$$= \left(\frac{n}{n-2}\right)^2 \widetilde{D}_0$$

$$- \frac{2}{(n-2)^2}\left(\left(\left|[x_k]_{\tilde{C}}\right| - \left|[x_k]_{\tilde{C}} \cap [x_k]_{\tilde{D}}\right|\right) - \left(m_{k,k} - min\left\{m_{k,k}, d_{k,k}\right\}\right)\right.$$

$$+ \left(\left|[x_{k+1}]_{\tilde{C}}\right| - \left|[x_{k+1}]_{\tilde{C}} \cap [x_{k+1}]_{\tilde{D}}\right|\right) - \left(m_{k+1,k} - min\left\{m_{k+1,k}, d_{k+1,k}\right\}\right)$$

$$\left. - \left(m_{k+1,k+1} - min\left\{m_{k+1,k+1}, d_{k+1,k+1}\right\}\right)\right)$$

Tính tương tự như vậy, ta được:

$$\widetilde{D}_s = \left(\frac{n}{n-s}\right)^2 \widetilde{D}_0 - \frac{2}{(n-2)^2}\sum_{i=0}^{s-1}\left(\frac{(|[x_{k+i}]_{\tilde{C}}| - |[x_{k+i}]_{\tilde{C}} \cap [x_{k+i}]_{\tilde{D}}|)}{-\sum_{j=0}^{i}(m_{k+1,k+j} - min\{m_{k+1,k+j}, d_{k+i,k+j}\})}\right)$$

Vì vậy,

$$\widetilde{D}_s = \left(\frac{n}{n-1}\right)^2 \widetilde{D}_0 - \frac{2}{(n-s)^2}\sum_{i=0}^{s-1}((|[x_{k+i}]_{\tilde{C}}| - |[x_{k+i}]_{\tilde{C}} \cap [x_{k+i}]_{\tilde{D}}|) - \varepsilon_i)$$

Với $\varepsilon_i = \sum_{j=0}^{i}\left(m_{k+i,k+j} - \min\left\{m_{k+i,k+j}, d_{k+i,k+j}\right\}\right)$

Thuật toán fifter để cập nhật tập rút gọn khi loại bỏ tập đối tượng:

Mệnh đề 7. *Cho BQĐ* $DS = (U, C \cup D)$*với* $U = \{x_1, x_2, \ldots, x_n\}$*và* $\tilde{R}$ *là một QHTĐM xác định trên miền giá trị của tập thuộc tính điều kiện.*$B \subseteq C$ *là TRG dựa trên khoảng cách mờ. Giả sử TĐT gồm s phần tử* $\Delta U = \{x_k, x_{k+1}, \ldots, x_{k+s-1}\}$ *bị loại khỏi* $U, s < n$*. Khi đó ta có:*

1) Nếu $D(x_{k+i}) = d$ với $i = 0, \ldots, s-1$ thì

$$\widetilde{D}_{U-\Delta U}\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right) = \left(\frac{n}{n-s}\right)^2 \widetilde{D}_U\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right) + \frac{2}{(n-s)^2}\sum_{i=0}^{s-1}|[x_{k+i}]_{\tilde{C}}| - |[x_{k+i}]_{\tilde{C}} \cap [x_{k+i}]_{\tilde{D}}|$$

2) Nếu $[x_{k+i}]_{\tilde{B}} \subseteq [x_{k+i}]_{\tilde{D}}$ với$i = 0, \ldots, s-1$ thì .

$$\widetilde{D}_{U-\Delta U}\left(\widetilde{\Upsilon_B^{\alpha}}, \widetilde{\Upsilon_{B\cup D}^{\alpha}}\right) = \tilde{R}_{U-\Delta U}\left(\widetilde{\Upsilon_C^{\alpha}}, \widetilde{\Upsilon_{C\cup D}^{\alpha}}\right)$$

Algorithm IF_FDAR_DelObj_α

Input: Đầu vào

1. Bảng quyết định $DS = (U, C \cup D)$ với $U = \{x_1, x_2, \ldots, x_n\}$, một QHTĐM $\tilde{Y}$, tập rút gọn $B \subseteq C$;
2. Ma trận tương đương mờ

$$M_U(\tilde{Y}_B) = [m_{ij}^B]_{n\times n}, M_U(\tilde{Y}_C) = [m_{ij}^C]_{n\times n}, M_U(\widetilde{\Upsilon_C^\alpha}) = [d_{ij}]_{n\times n}$$

3. Tập đối tượng gồm s phần tử bị loại bỏ $\Delta U = \{x_{k+1}, x_{k+2}, \ldots, x_{k+s-1}\}, s < n$

Output: Tập rút gọn xấp xỉ B_{best}của $DS' = (U - \Delta U, C \cup D)$ có độ chính xác phân lớp cao nhất.

1. $T := \emptyset$;
2. Đặt $X := \Delta U$;
3. For $i = 0$ to $s - 1$ do
4. If $[x_{k+i}]_{\tilde{B}} \subseteq [x_{k+i}]_{\tilde{D}}$ then $X := X - \{x_{k+i}\}$;
5. If $X = \emptyset$ then ReturnB_0;
6. Đặt $\Delta U := X$; $s = |\Delta U|$;
7. Tính các FPDs ban đầu:

$$\widetilde{D}_U(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha});\ \widetilde{D}_U(\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha})$$

8. Tính khoảng cách mờ bởi Mệnh đề 7 khi loại tập đối tượngΔU :

$$\widetilde{D}_{U-\Delta U}(\widetilde{\Upsilon_B^\alpha}, \Phi\widetilde{\Upsilon_{B\cup D}^\alpha});\ \widetilde{D}_{U-\Delta U}(\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha});$$

// **Giai đoạn Fifter, tìm các ứng viên cho tập rút gọn**

9. While $\widetilde{D}_{U-\Delta U}(\widetilde{\Upsilon_B^\alpha}, \Phi\widetilde{\Upsilon_{B\cup D}^\alpha}) \neq \widetilde{D}_{U-\Delta U}(\widetilde{\Upsilon_C^\alpha}, \widetilde{\Upsilon_{C\cup D}^\alpha})$ do
10. Begin
11. For each $a \in B$ do
12. Begin
13. Tính $\widetilde{D}_{U-\Delta U}\left((\widetilde{\Upsilon_{B-\{a\}}^\alpha}), \Upsilon_{B-\{a\}\cup D}^{\alpha}\right)$ bởi Mệnh đề 3.6 khi loại bỏ tập đối tượng ΔU;
14. Tính $SIG_{B-\{a\}}(a) := \widetilde{D}_{U-\Delta U}\left((\widetilde{\Upsilon_{B-\{a\}}^\alpha}), \Upsilon_{B-\{a\}\cup D}^{\alpha}\right) - \widetilde{D}_{U-\Delta U}(\widetilde{\Upsilon_B^\alpha}, \widetilde{\Upsilon_{B\cup D}^\alpha})$;
15. End;

16. Chọn $a_m \in B$ sao cho $SIG_B(a_m) = \underset{a \in B}{Min}\{SIG_{B-\{a\}}(a)\}$;
17. $B := B - \{a_m\}$;
18. $B_0 := B_0 - \{a_m\}$;
19. $T := T \cup B_0$;
20. End;

Độ phức tập của thuật toán IF_FDAR_DelObj_α được tính như bên dưới. Giả sử $D = \{d\}$. Độ phức tạp của vòng lặp trong câu lệnh 3 (For) là $O(|U| * |\Delta U|)$.

Trong trường hợp tốt nhất, thuật toán kết thúc ở câu lệnh 5 (khi TRG không thay đổi). Độ phức tạp của thuật toán IF_FDAR_DelObj_α là $O(|U| * |\Delta U|)$. Ngược lại, độ phức tạp của thuật toán tính khoảng cách mờ ở câu lệnh 7 là $O(|U|)$. Để tính độ phức tạp của thuật toán khi loại bỏ tập$|\Delta U|$ ra khỏi U ở câu lệnh 8, độ phức tạp là $O(|U| * |\Delta U|)$. Để tính giá trị của $SIG_B(a)$, ta phải tính$\widetilde{D}_{U-\Delta U}\left(\widetilde{(\Upsilon^{\alpha}_{B-\{a\}})}, \widetilde{\Upsilon^{\alpha}_{B-\{a\}\cup D}}\right)$. Độ phức tạp của $\widetilde{D}_{U-\Delta U}\left(\widetilde{(\Upsilon^{\alpha}_{B-\{a\}})}, \widetilde{\Upsilon^{\alpha}_{B-\{a\}\cup\{d\}}}\right)$ là $O(|U| * |\Delta U|)$. Do đó, độ phức tạp của vòng lặp While là $O(|B|^2 * |U| * |\Delta U|)$ và độ phức tạp của giai đoạn fifter trong trường hợp xấu nhất là $O(|B|^2 * |U| * |\Delta U|)$. Giả sử độ phức tạp của bộ phân lớp là $O(T)$ khi đó độ phức tạp của giai đoạn wrapper là $O\big((|B|) * T\big)$.

2.5. THUẬT TOÁN GIA TĂNG FILTER TÌM TẬP RÚT GỌN KHI BỔ SUNG TẬP THUỘC TÍNH

Cho BQĐ $DS = (U, C \cup D)$ với $U = \{x_1, x_2, \ldots, x_n\}$ khi đó, khoảng cách mờ giữa hai tập thuộc tính C và D theo Mệnh đề 2.3 được đề xuất trong Chương 2 được xác định như sau:

$$\widetilde{D}_{U-\Delta U}\left(\widetilde{\Upsilon^{\alpha}_{C}}, \widetilde{\Upsilon^{\alpha}_{C\cup D}}\right) = \frac{1}{n^2}\sum_{i=1}^{n}\left(|[x_i]_{\tilde{C}}| - |[x_i]_{\tilde{C}} \cap [x_i]_{\tilde{D}}|\right)$$

Mệnh đề 8. *Cho BQĐ $DS = (U, C \cup D)$ với $U = \{u_1, u_2, \ldots, u_n\}$. Giả sử tập thuộc tính điều kiện B được bổ sung vào C với $B \cap C = \emptyset$. Giả sử $M(\tilde{R}_B) =$*

$[b_{ij}]_{n\times n}$, $M(\tilde{R}_C) = [c_{ij}]_{n\times n}$, $M(\tilde{R}_D) = [d_{ij}]_{n\times n}$ *là các ma trận tương đương mờ của các QHTĐM* $\tilde{R}_B, \tilde{R}_C, \tilde{R}_D$ *trên B, C, D tương ứng. Khi đó ta có:*

1) Nếu $c_{ij} \le d_{ij}$ *với mọi* $1 \le i,j \le n$ *thì* $\tilde{D}(C \cup B, C \cup B \cup D) = 0$

2) Nếu $b_{ij} \ge c_{ij}$ *với mọi* $1 \le i,j \le n$ *thì*

$$\tilde{D}(C \cup B, C \cup B \cup D) = \tilde{D}(C, C \cup D) = \frac{1}{n^2}.\sum_{i=1}^{n}\sum_{j=1}^{n}(c_{ij} - min(c_{ij}, d_{ij}))$$

3) Nếu $b_{ij} < c_{ij}$ *với mọi* $1 \le i,j \le n$ *thì*

$$\tilde{D}(C \cup B, C \cup B \cup D) = \tilde{D}(B, B \cup D) = \frac{1}{n^2}.\sum_{i=1}^{n}\sum_{j=1}^{n}(b_{ij} - min(b_{ij}, d_{ij}))$$

Chứng minh: Khi bổ sung thêm B vào C, khoảng cách mờ được xác định như sau:

$$\tilde{D}(C \cup B, C \cup B \cup D) = \frac{1}{n^2}\sum_{i=1}^{n}(|[u_i]_{\widetilde{C\cup B}}| - |[u_i]_{\widetilde{C\cup B}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2}.\sum_{i=1}^{n}(|[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(min(c_{ij}, b_{ij}) - min(c_{ij}, b_{ij}, d_{ij}))$$

1) Nếu $c_{ij} \le d_{ij}$ với mọi $1 \le i,j \le n$ thì $[u_i]_{\tilde{C}} \subseteq [u_i]_{\tilde{D}}$ và $[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}} = [u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}}$. Từ đó ta có:

$$\tilde{D}(C \cup B, C \cup B \cup D) = \frac{1}{n^2}.\sum_{i=1}^{n}(|[u_i]_{\widetilde{C\cup B}}| - |[u_i]_{\widetilde{C\cup B}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2}.\sum_{i=1}^{n}(|[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}}|) = 0$$

2) Từ $b_{ij} \ge c_{ij}$ ta có $[u_i]_{\tilde{C}} \subseteq [u_i]_{\tilde{B}}$ và $[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} = [u_i]_{\tilde{C}}$ với mọi $u_i \in U$. Từ đó ta có:

$$\widetilde{D}(C \cup B, C \cup B \cup D) = \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2} \sum_{i=1}^{n} (|[u_i]_{\tilde{C}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{D}}|) = \widetilde{D}(C, C \cup \{d\})$$

$$= \frac{1}{n^2} \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} (c_{ij} - min(c_{ij}, d_{ij}))$$

3) Từ $b_{ij} < c_{ij}$ ta có $[u_i]_{\tilde{B}} \subset [u_i]_{\tilde{C}}$ và $[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} = [u_i]_{\tilde{B}}$ với mọi $u_i \in U$. Từ đó ta có:

$$\widetilde{D}(C \cup B, C \cup B \cup D) = \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2} \sum_{i=1}^{n} (|[u_i]_{\tilde{B}}| - |[u_i]_{\tilde{B}} \cap [u_i]_{\tilde{D}}|) = \widetilde{D}(B, B \cup D)$$

$$= \frac{1}{n^2} \cdot \sum_{i=1}^{n} \sum_{j=1}^{n} (b_{ij} - min(b_{ij}, d_{ij}))$$

Thuật toán filter tìm các ứng viên cho TRG mỗi khi bổ sung thuộc tính có độ quan trọng lớn nhất:

Thuật toán IF_FDAR_AA_α (Incremental Filter Fuzzy Distance-based Attribute Reduction Algorithm when Adding Attributes α).

Đầu vào:

1) Bảng quyết định $DS = (U, C \cup D)$ với $U = \{u_1, u_2, \ldots, u_n\}$, tập rút gọn $R \subseteq C$, các ma trận tương đương mờ $M(\tilde{R}_C) = [c_{ij}]_{n \times n}$, $M(\tilde{R}_D) = [d_{ij}]_{n \times n}$ của các QHTĐM $\tilde{R}_C, \tilde{R}_D$, khoảng cách mờ $\widetilde{D}(C, C \cup D)$;

2) Tập thuộc tính bổ sung B với $B \cap C = \emptyset$;

Đầu ra: Tập rút gọn R_1 của $DS_1 = (U, C \cup B \cup D)$

Bước 1: Khởi tạo và kiểm tra tập thuộc tính bổ sung

1. $T := \emptyset$; *// Chứa các ứng viên tập rút gọn*
2. Tính ma trận QHTĐM $M(\tilde{R}_B) = [b_{ij}]_{n\times n}$;
3. If $b_{ij} \geq c_{ij}$ với mọi $1 \leq i \leq n, 1 \leq j \leq n$ then Return R;
4. If $b_{ij} < c_{ij}$ với mọi $1 \leq i \leq n, 1 \leq j \leq n$ then $R = \emptyset$; *//Tìm tập rút gọn trong tập B*

Bước 2: Thực hiện thuật toán tìm tập rút gọn

// Giai đoạn filter, tìm các ứng viên cho tập rút gọn xuất phát từ tập R.

5. While $\tilde{D}(R, R \cup D) \neq \tilde{D}(C \cup B, C \cup B \cup D)$ do
6. Begin
7. For each $a \in B$ tính $SIG_R(a) = \tilde{D}(R, R \cup D) - \tilde{D}(R \cup \{a\}, R \cup \{a\} \cup D)$ với $\tilde{D}(R \cup \{a\}, R \cup \{a\} \cup D)$ được tính bởi công thức trong Mệnh đề 3.7.
8. Chọn $a_m \in B$ sao cho $SIG_R(a_m) = \underset{a\in B}{Max}\{SIG_R(a)\}$;
9. $R := R \cup \{a_m\}$;
10. $T := T \cup R$;
11. End;

2.6. THUẬT TOÁN GIA TĂNG FILTER TÌM TẬP RÚT GỌN KHI LOẠI BỎ TẬP THUỘC TÍNH

Mệnh đề 9. *Cho BQĐ* $DS = (U, C \cup D)$ *với* $U = \{u_1, u_2, \ldots, u_n\}$. *Giả sử tập thuộc tính điều kiện B được loại bỏ khỏi C với* $B \subset C$ *và* $A = C - B$ *là tập thuộc tính còn lại. Đặt* $M(\tilde{R}_B) = [b_{ij}]_{n\times n}$, $M(\tilde{R}_C) = [c_{ij}]_{n\times n}$, $M(\tilde{R}_A) = [a_{ij}]_{n\times n}$, $M(\tilde{R}_D) = [d_{ij}]_{n\times n}$ *tương ứng là ma trận tương đương mờ của các QHTĐM* $\tilde{R}_B, \tilde{R}_C, \tilde{R}_A, \tilde{R}_D$. *Khi đó ta có:* $\tilde{D}(A, A \cup \{d\}) = \tilde{D}(C, C \cup \{d\}) + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}[a_{ij} - c_{ij} + min(c_{ij}, d_{ij}) - min(a_{ij}, d_{ij})]$

Chứng minh: Ta có:

$$\widetilde{D}(A, A \cup D) = \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{A}}| - |[u_i]_{\tilde{A}} \cap [u_i]_{\tilde{D}}|)$$

$$= \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{C}}| - |[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{D}}|) + \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{A}} - [u_i]_{\tilde{C}}|)$$

$$+ \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{C}} \cap [u_i]_{\tilde{D}}|) - \frac{1}{n^2} \cdot \sum_{i=1}^{n} (|[u_i]_{\tilde{A}} \cap [u_i]_{\tilde{D}}|)$$

$$= \widetilde{D}(C, C \cup \{d\}) + \frac{1}{n^2} \cdot \sum_{i=1}^{n} (a_{ij} - c_{ij}) + \frac{1}{n^2} \cdot \sum_{i=1}^{n} (min(c_{ij}, d_{ij}))$$

$$- \frac{1}{n^2} \cdot \sum_{i=1}^{n} (min(a_{ij}, d_{ij}))$$

$$= \widetilde{D}(C, C \cup \{d\}) + \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} [a_{ij} - c_{ij} + min(c_{ij}, d_{ij}) - min(a_{ij}, d_{ij})]$$

Thuật toán gia tăng filter tìm TRG trong BQĐ sử dụng khoảng cách mờ khi loại bỏ tập thuộc tính B như sau:

Thuật toán IF_FDAR_DA_α (Incremental Filter Fuzzy Distance-based Attribute Reduction Algorithm when Deleting Attributes α).

Đầu vào:

1) Bảng quyết định $DS = (U, C \cup D)$ với $U = \{u_1, u_2, \ldots, u_n\}$, tập rút gọn $R \subseteq C$, các ma trận tương đương mờ $M(\tilde{R}_C) = [c_{ij}]_{n \times n}$, $M(\tilde{R}_D) = [d_{ij}]_{n \times n}$, khoảng cách mờ $\widetilde{D}(C, C \cup D)$;

2) Tập thuộc tính B loại bỏ khỏi C với $B \subset C$;

Đầu ra: Tập rút gọn R_1 của $DS_1 = (U, (C - B) \cup D)$;

1) ***Trường hợp 1:*** If $B \subseteq C - R$ then Retturn (R);

2) ***Trường hợp 2:*** If $R \subseteq B$ then thực hiện thuật toán không gia tăng

filter-wrapper tìm tập rút gọn sử dụng khoảng cách FW_FDBAR_α trong mục 2.2 Chương 2.

3) Trường hợp 3: If $R \cap B \neq \emptyset$ then thực hiện các bước của thuật toán tìm tập rút gọn.

Bước 1: Khởi tạo

1. Đặt $T := \emptyset$; $A := C - B$; *// Chứa các ứng viên tập rút gọn*
2. Tính ma trận tương đương mờ $M(\tilde{R}_B) = [b_{ij}]_{n\times n}$, $M(\tilde{R}_A) = [a_{ij}]_{n\times n}$
3. Đặt $R := R - B$ *//Xét các thuộc tính trong tập rút gọn*

Bước 2: Thực hiện thuật toán tìm tập rút gọn

// Giai đoạn filter, tìm các ứng viên cho tập rút gọn xuất phát từ tập R.

4. While $\tilde{D}(R, R \cup D) \neq \tilde{D}(A, A \cup D)$ do
5. Begin
6. For each $a \in R$ tính $SIG_R(a) = D\big(R - \{a\}, \{R - \{a\}\} \cup D\big) - D(R, R \cup D)$ với $\tilde{D}\big(R - \{a\}, \{R - \{a\}\} \cup D\big)$;
7. Chọn $a_m \in R$ sao cho $SIG_R(a_m) = \underset{a\in R}{Min}\{SIG_R(a)\}$;
8. $R := R - \{a_m\}$;
9. $T := T \cup R$;
10. End;

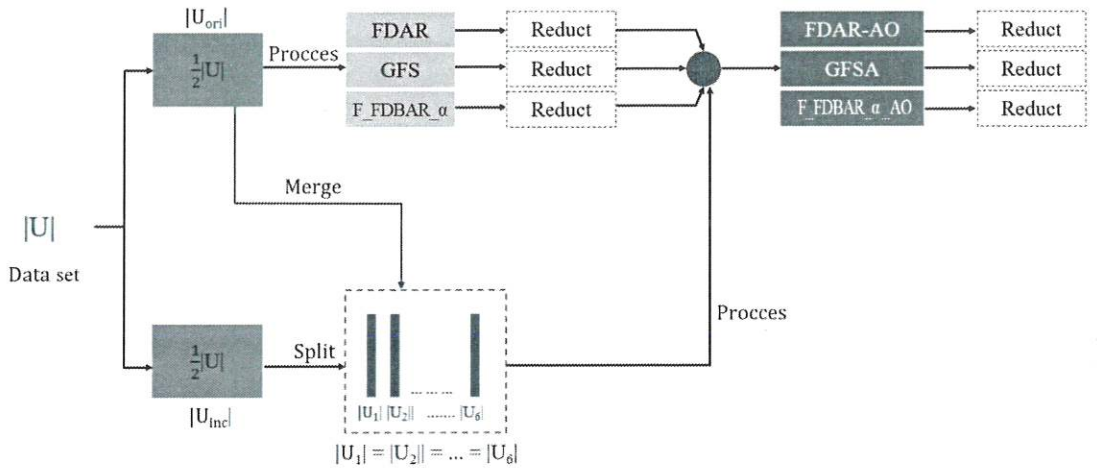
CHƯƠNG 3. QUÁ TRÌNH THỰC NGHIỆM VÀ KẾT QUẢ

3.1. SO SÁNH CÁC THUẬT TOÁN TRÊN BẢNG QUYẾT ĐỊNH KHI BỔ SUNG TẬP ĐỐI TƯỢNG

Quy trình thực nghiệm trong phần này sẽ được trình bày như hình 3.1. Đầu tiên, tập dữ liệu sẽ được chia thành hai phần bằng nhau. Phần thứ nhất ký hiệu là U_{ori} (cột (4) trong bảng 3.1) được sử dụng cho thuật toán F_FDBAR_α, FDAR và GFS nhằm tìm kiếm một rút gọn và phần thứ hai ký hiệu là U_{inc} (cột (5) bảng 3.1) được sử dụng cho các thuật toán gia tăng. Tiếp theo, TĐT gia tăng U_{inc} sẽ được chia tiếp thành 6 phần bằng nhau U_1, U_2, U_3, U_4, U_5, U_6 để bổ sung lần lượt vào BQĐ. Các thuật toán gia tăng sau đó sẽ được tính toán lần lượt trên các bộ dữ liệu này nhằm đánh giá các kết quả thu được. Trong bảng 3.1, các cột $|U|$, $|U_{ori}|$, $|U_{inc}|$, $|C|$, $|D|$ ký hiệu lần lượt cho số lượng các đối tượng trong tập dữ liệu, số lượng đối tượng trong tập U_{ori}, số lượng đối tượng trong tập U_{inc}, số lượng thuộc tính điều kiện và số lớp quyết định.

Bảng 3.1: Các bộ dữ liệu sử dụng trong thử nghiệm

ID	Data sets	|U|	$\|U_{ori}\|$	$\|U_{inc}\|$	|C|	|D|
1	Ionosphere	351	175	176	34	2
2	Leaf	340	170	170	15	30
3	Movement-libras	360	180	180	90	15
4	Wall	5456	2728	2728	24	4



Hình 3.1: ***Quy trình thực nghiệm các thuật toán gia tăng bổ sung đối tượng***

Như đã đề cập, trước hết chúng tôi sẽ so sánh hiệu quả của các thuật toán FDAR trong [9], GFS trong [6] với thuật toán đề xuất. Đây đều là các thuật toán được xử lý trên BQĐ khi chưa có sử bổ sung hay loại bỏ đối tượng. Mục đích so sánh thuật toán đề xuất với hai thuật toán này để làm rõ hiệu quả của cách tiếp cận theo lát cắt α trong vấn đề tìm TRG trên các bảng dữ liệu nhiễu hay mang các thông tin không chắc chắn. Kích thước và độ chính xác phân lớp của các TRG được trình bày chi tiết tại bảng 3.2. Trong đó, ký hiệu |**C**| là số thuộc tính của bộ dữ liệu ban đầu, |**B**| là số thuộc tính của tập rút gọn, **Acc** trình bày độ chính xác phân lớp và **time** là thời gian tính toán được tính bằng đơn vị "giây". Trên hầu hết tất cả các tập dữ liệu, có thể thấy rằng TRG thu được từ thuật toán F_FDBAR_α có kích thước(|B|) nhỏ nhất, trong khi đó, các TRG thu được từ FDAR vẫn có kích thước rất lớn và chưa tối ưu. Ví dụ, các bộ dữ liệu wall, ionosphere, kích thước TRG thu được trên các thuật toán FDAR và GFS vẫn còn rất lớn, trong khi kích thước thu được từ thuật toán F_FDBAR_α là rất nhỏ.

Tiếp theo chúng tôi so sánh thời gian tính toán giữa ba thuật toán. Thời gian của các thuật toán được tính sau bước tiền xử lý dữ liệu tới khi các rút gọn được xác định. Dựa trên kết quả của bảng 3.2, thời gian của thuật toán GFS là nhanh hơn các thuật toán còn lại trên toàn bộ các tập dữ liệu. Điều này có thể được lý giải đó là các thuật toán dựa trên không gian tập mờ và tập mờ trực cảm phải tính toán các ma trận

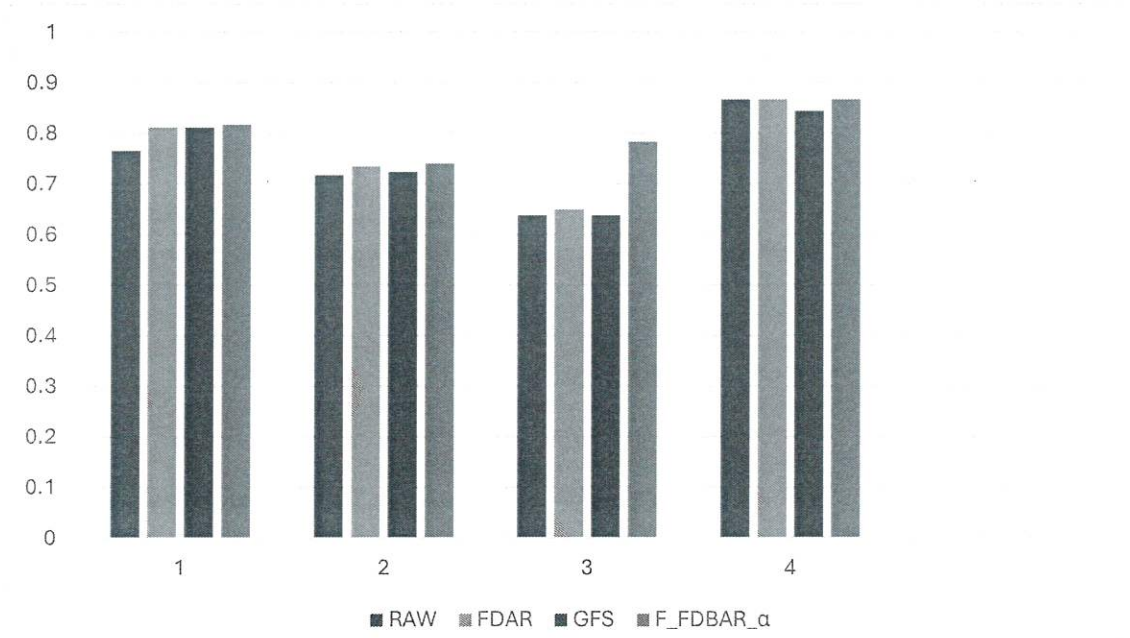
quan hệ có nhiều phần tử. Ngoài ra, thuật toán FDAR chỉ sử dụng độ tương tự để tính toán các độ đo, còn thuật toán F_FDBAR_α phải sử dụng cả độ tương tự và độ khác biệt để thực hiện tính toán. Do đó, thuật toán F_FDBAR_α sẽ có độ phức tạp tính toán là lớn nhất. Chúng tôi tiếp tục so sánh độ chính xác phân lớp của ba thuật toán thông qua bộ phân lớp KNN. Bảng 3.3, hình 3.2 và hình 3.3 trình bày chi tiết các kết quả so sánh của ba thuật toán trên các bộ dữ liệu. Trong đó, cột RAW là độ chính xác phân lớp gốc được sử dụng toàn bộ thuộc tính của từng tập dữ liệu để đánh giá. Có thể nhận thấy, phương pháp của chúng tôi xác định được các thuộc tính quan trọng rất hiệu quả cho các tập dữ liệu khác nhau. Đặc biệt hơn, khi so sánh với dữ liệu gốc, hiệu quả của các rút gọn từ thuật toán đề xuất vượt trội hơn trong 4 trường hợp.

Bảng 3.2: ***Kết quả xử lý của FDAR, GFS và* F_FDBAR_α *trên* |*Uori*|**

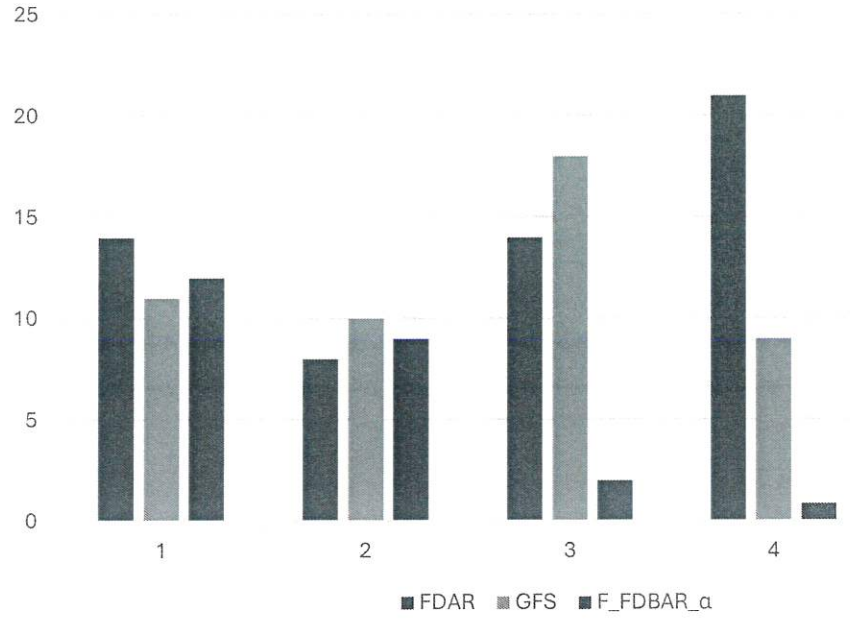
ID	Data sets	RAW	FDAR			GFS			F_FDBAR_α		
		Acc	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time
1	**ionosphere**	0.766± 0.092	14	0.812 ±0.113	0.035	11	0.812 ±0.085	0.013	**12**	**0.818 ±0.123**	0.02
2	**leaf**	0.718± 0.101	8	0.735 ±0.121	0.005	10	0.724 ±0.121	0.016	**9**	**0.741 ±0.096**	0.006
3	**wall**	0.638 ± 0.094	14	0.650 ±0.076	5.597	18	0.638 ±0.094	6.997	**2**	**0.784 ±0.059**	0.911
4	**movement_libras**	0.867 ± 0.090	21	0.867±0.097	0.091	9	0.844 ± 0.096	0.011	**10**	**0.867 ±0.071**	0.05
	AVG	0.747 ±0.094	**15**	0.766 ±0.102	1.443	12	0.755 ± 0.099	**1.759**	8.25	**0.803 ±0.087**	0.247

Trong phần này, chúng tôi sẽ trình bày các kết quả so sánh từ ba phương pháp gia tăng F_FDBAR_α _AO, FDAR_AO và IFSA. Trong đó, đầu vào của ba phương pháp này gồm các rút gọn được tính toán từ các phương pháp tương ứng ARIFPD, FDAR và GFS. Một điều hiển nhiên là các thuật toán gia tăng có thời gian xử lý

nhanh hơn rất nhiều so với các thuật toán FDAR, IFSA và F_FDBAR_α vì các thuật toán này chỉ tính toán trên các phần bổ sung của bảng dữ liệu, thay vì toàn bộ bảng dữ liệu. Bảng 3.2 cho thấy rằng đối với hầu hết các tập dữ liệu, thời gian thực hiện của FDAR_AO và IFSA nhanh hơn khi so sánh với F_FDBAR_α _AO. Điều này có thể được giải thích tương tự như khi chúng ta so sánh thời gian thực hiện của thuật toán FDAR và F_FDBAR_α. Hơn nữa, thuật toán đề xuất bao gồm một bước xử lý để loại bỏ các thuộc tính dư thừa. Do đó thời gian xử lý thuật toán của chúng tôi sẽ chậm hơn. Tuy nhiên, thời gian thực hiện của ARIFPD_AO tốt hơn FDAR_AO trên một số bộ dữ liệu. Điều này là do kích thước TRG thu được từ thuật toán đề xuất nhỏ hơn hai thuật toán còn lại, khi đó số vòng lặp được tiến hành ít hơn.



Hình 3.2: Độ chính xác phân lớp của các thuật toán



Hình 3.3: Kích thước tập rút gọn của các thuật toán

Bảng 3.3: Kết quả xử lý của FDAR_AO, GFS VÀ F_FDBAR_α _AO

ID	Adding data sets	RAW	FDAR_AO			IFSA			F_FDBAR_α _AO		
		Acc	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time
1	$\|U_1\|$ = 210	0.776±0.102	19	0.800±0.108	0.006	10	0.824±0.117	0.002	**12**	**0.819±0.131**	0
	$\|U_2\|$ = 245	0.792±0.098	20	0.821±0.100	0.001	12	0.825±0.111	0.006	**15**	**0.837±0.112**	0.005
	$\|U_3\|$ = 280	0.807±0.070	21	0.832 ± .078	0.001	12	0.811±0.085	0.001	**15**	**0.854±0.081**	0
	$\|U_4\|$ = 315	0.832±0.059	21	0.848±0.058	0.001	14	0.836±0.091	0.004	**15**	**0.858±0.061**	0
	$\|U_5\|$ = 351	0.838±0.064	21	0.852±0.055	0.001	16	0.849±0.057	0.008	**15**	**0.875±0.069**	0
2	$\|U_1\|$ = 204	0.683±0.109	10	0.692±0.074	0.017	11	0.683±0.105	0.001	**10**	**0.692±0.074**	0.001
	$\|U_2\|$ = 238	0.643±0.114	10	0.643±0.082	0.002	12	0.656±0.112	0.001	**11**	**0.677±0.095**	0
	$\|U_3\|$ = 272	0.608±0.118	11	0.622±0.106	0.001	12	0.608±0.019	0.001	**11**	**0.622±0.106**	0
	$\|U_4\|$ = 306	0.588±0.074	11	0.588±0.069	0.001	12	0.602±0.067	0.001	**11**	**0.588±0.069**	0
	$\|U_5\|$ = 340	0.606±0.063	12	0.612±0.055	0.002	14	0.612±0.068	0.003	**12**	**0.612±0.055**	0.001
3	$\|U_1\|$ = 3273	0.690±0.048	17	0.676±0.027	0.522	21	0.681±0.048	0.643	**2**	**0.787±0.035**	0

	$\|U_2\|$ = 3818	0.729±0.083	18	0.725±0.063	0.153	22	0.712±0.087	0.210	**2**	**0.806±0.05**	0
	$\|U_3\|$ = 4363	0.755±0.064	18	0.747±0.065	0.001	23	0.745±0.064	0.308	**2**	**0.801±0.031**	0
	$\|U_4\|$ = 4908	0.766±0.071	18	0.770±0.068	0.001	23	0.763±0.072	0.001	**2**	**0.818±0.022**	0
	$\|U_5\|$ = 5456	0.773±0.059	18	0.770±0.066	0.001	23	0.773±0.060	0.001	**2**	**0.819±0.034**	0
4	$\|U_1\|$ = 216	0.851±0.105	29	0.847±0.102	0.032	9	0.833±0.104	0.001	**28**	**0.828±0.102**	0.076
	$\|U_2\|$ = 252	0.770±0.101	30	0.762±0.104	0.004	11	0.750±0.069	0.003	**28**	**0.771±0.101**	0
	$\|U_3\|$ = 288	0.767±0.102	31	0.750±0.100	0.004	14	0.729±0.105	0.006	**28**	**0.753±0.112**	0
	$\|U_4\|$ = 324	0.739±0.110	31	0.723±0.104	0.001	14	0.733±0.106	0.001	**28**	**0.721±0.115**	0
	$\|U_5\|$ = 360	0.758±0.117	31	0.736±0.110	0.001	14	0.750±0.112	0.001	28	**0.739±0.129**	0

Độ chính xác và kích thước của các TRG được xác định bằng phương pháp của đề án cũng được trình bày trong bảng 3.3. Kích thước của TRG trong mỗi giai đoạn tăng dần, kích thước TRG của F_FDBAR_α _AO nhỏ hơn nhiều so với FDAR_AO và IFSA, đặc biệt đối với một số tập dữ liệu có số lượng thuộc tính lớn. Nói cách khác, các phương pháp RGTT dựa trên cách tiếp cận tập thô và các phần mở rộng của nó gặp nhiều khó khăn trong việc nâng cao độ chính xác phân loại cho dữ liệu nhiễu.

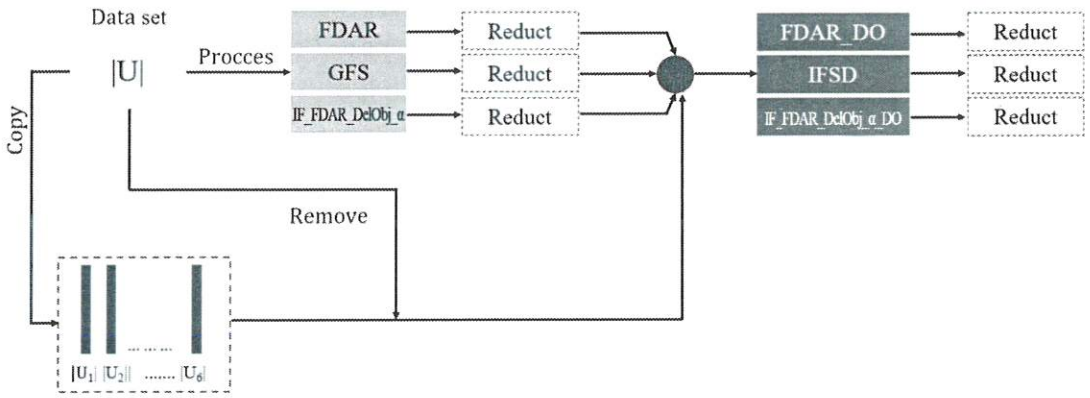
3.2. SO SÁNH CÁC THUẬT TOÁN TRÊN BẢNG QUYẾT ĐỊNH KHI LOẠI BỎ TẬP ĐỐI TƯỢNG

Để thực hiện các đánh giá trên các thuật toán gia tăng khi loại bỏ TĐT, dữ liệu thử nghiệm được chúng tôi trình bày trong bảng 3.4. Tập U_{dec} được phân chia tiếp thành 6 phần bằng nhau (biểu diễn từ U_1 đến U_6) được sử dụng để làm các TĐT bị loại bỏ. Trong bảng 3.4, các cột $|U_{dec}|$, $|C|$, $|D|$ được ký hiệu lần lượt cho tổng số đối tượng hay bản ghi trong U_{dec}, số lượng thuộc tính điều kiện và số lượng lớp quyết định.

Bảng 3.4: ***Các bộ dữ liệu sử dụng trong thử nghiệm***

ID	Data sets	$\|U\|$	$\|U_{dec}\|$	$\|C\|$	$\|D\|$
1	Robot-failures	164	84	90	5
2	Ionosphere	351	175	34	2
3	Movement-libras	360	180	90	15
4	Wall	5456	2728	24	4

Quy trình thực nghiệm được mô tả như hình 3.4. Đầu tiên, tập dữ liệu sẽ được sao chép thành một tập dữ liệu khác. Ở tập dữ liệu được sao chép chúng tôi chia thành 6 phần bằng nhau tương ứng với các phần dữ liệu từ U_1 đến U_6. Cũng tương tự như phần 3.1, báo cáo tiếp tục sử dụng các thuật toán IF_FDAR_DelObj_α, GFS và FDAR để tìm TRG trên toàn bộ U. Sau đó, sử dụng các thuật toán FDAR_DO, IFSD và IF_FDAR_DelObj_α _DO để tìm tiếp các TRG khi tập dữ liệu trên U loại bỏ lần lượt các phần từ U_1 đến U_6.



Hình 3.4: ***Quy trình thử nghiệm các thuật toán gia tăng loại bỏ đối tượng***

Kích thước TRG cũng như hiệu quả phân lớp của ba phương pháp được trình bày trong bảng 3.4. Cũng giống như các kết quả trong bảng 3.2, kích thước TRG thu được từ IF_FDAR_DelObj_α vẫn là nhỏ nhất và có độ chính xác phân lớp cao nhất trên phần lớp các bộ dữ liệu như Robot-failures, Ionosphere, Wall. Khi so sánh với dữ liệu gốc, độ chính xác trong mô hình phân lớp KNN của thuật toán đề xuất cũng thể hiện khả năng vượt trội.Ngoài ra, kích thước tập rút gọn từ thuật toánIF_FDAR_DelObj_α cao hơn so với tập rút gọn trung bình của dữ liệu gốc và cao nhất trong ba thuật toán mặc dù số lượng thuộc tính hay kích thước TRG thu được cũng là nhỏ nhất.

Bảng 3.5: ***Kết quả xử lý của FDAR, GFS và* IF_FDAR_DELOBJ_α *trên U***

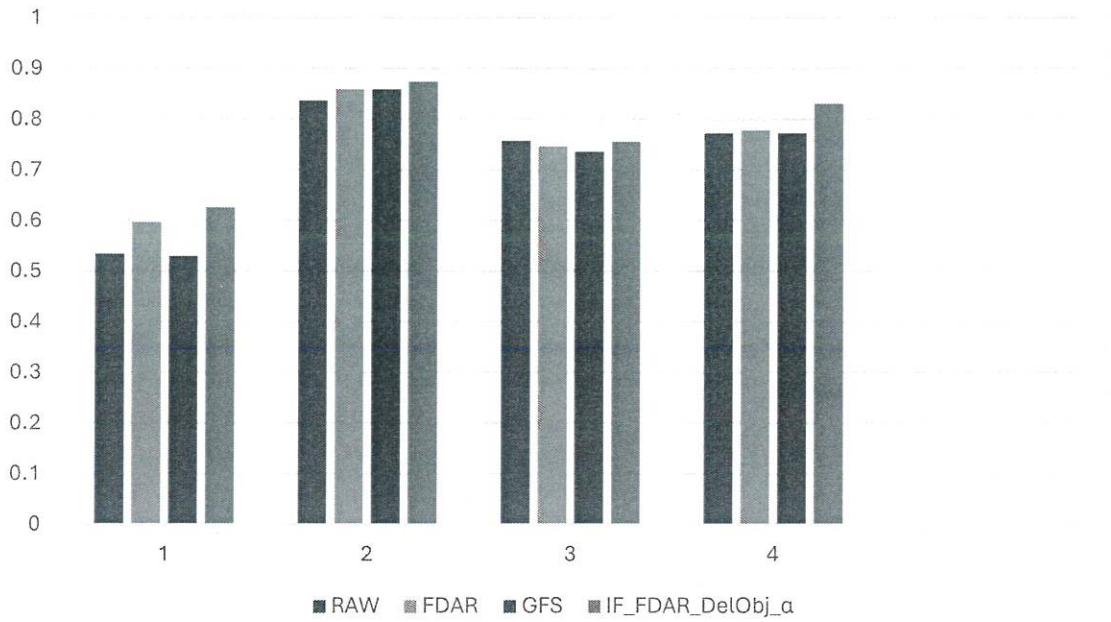
ID	RAW	FDAR			GFS			IF_FDAR_DelObj_α		
	Acc	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time
1	0.535 ±0.099	19	0.597 ±0.097	0.071	18	0.53±0.137	0.007	**15**	0.627±0.119	0.06
2	0.838 ±0.064	14	0.860 ±0.049	0.058	14	0.860±0.079	0.016	**15**	0.875±0.048	0.066
3	0.758 ±0.117	20	0.747 ±0.126	0.248	25	0.736±0.150	0.057	**17**	0.756±0.116	0.225
4	0.773	15	0.779	23.32	23	0.773±0.060	15.56	**4**	0.831±0.04	7.606

	±0.059		±0.074							
AVG	0.726 ±0.085	**17**	0.746 ±0.087	**5.924**	20	0.725±0.107	3.91	12.75	**0.772±0.081**	1.990

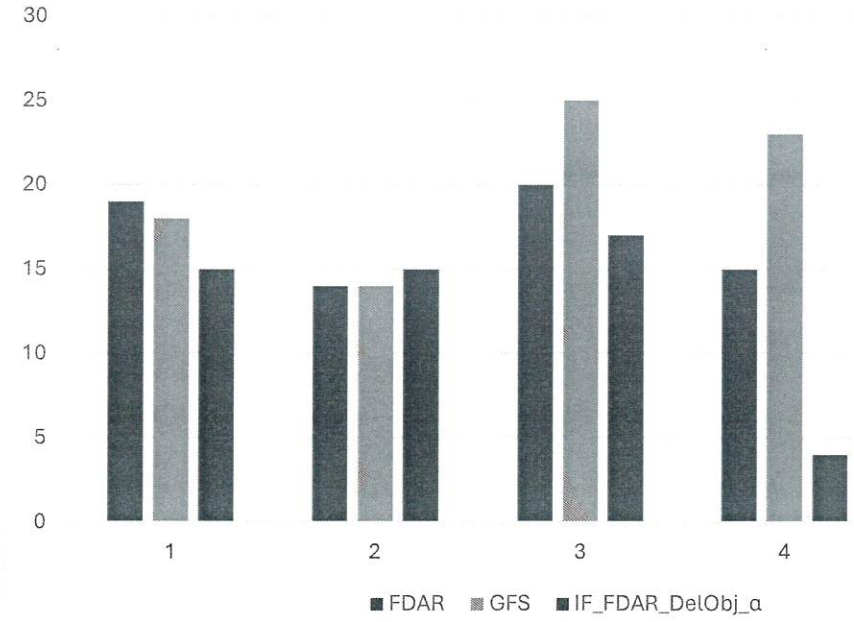
Bảng 3.6: ***Kết quả xử lý của FDAR_DO, GFS VÀ* IF_FDAR_DELOBJ_α_*DO***

ID	Adding data sets	RAW	FDAR_DO			IFSD			IF_FDAR_DELOBJ_α_*DO*		
		Acc	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time	$\|B\|$	Acc	Time
1	$\|U_1\|$ = 148	0.573±0.102	18	0.635 ±0.129	0.007	17	0.568±0. 095	0.004	**14**	0.64±0. 169	0.06
	$\|U_2\|$ = 132	0.575±0.107	17	0.605 ±0.068	0.006	17	0.568±0. 094	0.003	**13**	0.605± 0.093	0.004
	$\|U_3\|$ = 116	0.588±0.104	17	0.605 ±0.092	0.006	16	0.630±0. 089	0.003	**12**	0.639± 0.122	0.004
	$\|U_4\|$ = 100	0.660±0.102	17	0.670+0 .078	0.005	15	0.700±0. 110	0.002	**12**	0.68±0. 087	0.003
	$\|U_5\|$ = 84	0.629±0.105	16	0.665 ±0.110	0.005	13	0.665±0. 142	0.002	**12**	0.675± 0.101	0.003
2	$\|U_1\|$ = 316	0.830±0.057	14	0.852 ±0.048	0.011	14	0.855 ±0.056	0.008	**15**	0.865± 0.068	0.011
	$\|U_2\|$ = 281	0.808±0.070	14	0.851 ±0.065	0.011	12	0.847 ±0.078	0.006	**15**	0.854± 0.082	0.01
	$\|U_3\|$ = 252	0.794±0.095	13	0.846 ±0.085	0.012	11	0.834 ±0.108	0.006	**15**	0.867± 0.104	0.011
	$\|U_4\|$ = 216	0.777±0.109	13	0.839 ±0.098	0.009	10	0.806 ±0.130	0.005	**15**	0.825± 0.131	0.012
	$\|U_5\|$ = 180	0.773±0.097	13	0.819 ±0.089	0.009	10	0.790 ±0.088	0.004	**15**	0.795± 0.106	0.011

3	$\|U_1\|$ = 324	0.739±0.110	20	0.727 ±0.119	0.021	22	0.724 ±0.131	0.013	**17**	0.742± 0.12	0.016
	$\|U_2\|$ = 288	0.767±0.102	19	0.747 ±0.113	0.019	20	0.754 ±0.110	0.011	**17**	0.771± 0.107	0.014
	$\|U_3\|$ = 252	0.770±0.101	18	0.755 ±0.111	0.019	17	0.751 ±0.090	0.009	**16**	0.778± 0.094	0.016
	$\|U_4\|$ = 216	0.851±0.105	17	0.847 ±0.088	0.018	16	0.833 ±0.090	0.007	**16**	0.865± 0.085	0.014
	$\|U_5\|$ = 180	0.867±0.090	16	0.861 ±0.094	0.013	14	0.811 ±0.090	0.005	**15**	0.878± 0.074	0.012
4	$\|U_1\|$ = 4911	0.765±0.070	14	0.782 ±0.070	6.825	23	0.763 ±0.072	1.123	**3**	0.85± 0.03	0.388
	$\|U_2\|$ = 4366	0.755±0.065	13	0.776 ±0.068	5.991	23	0.746 ±0.066	0.914	**2**	0.774± 0.034	0.202
	$\|U_3\|$ = 3821	0.728±0.082	12	0.750 ±0.076	5.103	22	0.712 ±0.082	0.746	**2**	0.764± 0.036	0
	$\|U_4\|$ = 3276	0.693±0.044	11	0.715 ±0.050	4.345	20	0.690 ±0.042	0.626	**2**	0.766± 0.054	0.001
	$\|U_5\|$ = 2731	0.633±0.096	10	0.654 ±0.078	3.551	19	0.639 ±0.093	0.505	**2**	0.733± 0.069	0



Hình 3.5: Độ chính xác phân lớp của các thuật toán IF_FDAR_DelObj_α



Hình 3.6: Kích thước tập rút gọn của các thuật toán IF_FDAR_DelObj_α

KẾT LUẬN

1. Các kết quả đạt được của luận án

Với mục tiêu chính là giảm số lượng đặc trưng và nâng cao khả năng phân loại, giảm thuộc tính được coi là một vấn đề quan trọng trong quá trình tiền xử lý dữ liệu. Trong nghiên cứu này, tác giả đề xuất một phương pháp đo lường cho khoảng cách phân vùng tập mờ và xây dựng một công thức tăng cường để cập nhật khoảng cách phân vùng tập mờ khi thêm một bộ đối tượng.

Dựa trên đó, tác giả phát triển thuật toán dựa trên phương pháp tập mờ . Thuật toán đầu tiên được thiết kế để tìm ra tập thuộc tính giảm trên bảng quyết định khi không có bộ đối tượng bổ sung. Thuật toán thứ hai là một thuật toán tăng cường, nhằm tìm ra tập thuộc tính giảm xấp xỉ khi bảng quyết định có sự gia tăng về tập đối tượng. So với các phương pháp dựa trên các phương pháp tập mờ và tập mờ trực giác, kết quả thực nghiệm chứng minh rằng phương pháp của chúng tôi có khả năng cải thiện độ chính xác trên các bộ dữ liệu không nhất quán hoặc có độ chính xác phân loại ban đầu thấp.

2. Định hướng phát triển

(1) Triển khai các thuật toán đề xuất vào việc giải quyết các lớp bài toán trong thực tiễn, đặc biệt các bài toán có dữ liệu với số thuộc tính lớn (high dimention data) trong các lĩnh vực khác nhau như dữ liệu gen trong tin sinh học…

(2) Tiếp tục nghiên cứu, đề xuất các thuật toán gia tăng hiệu quả nhằm giảm thiểu thời gian thực hiện dựa trên các mô hình tập thô mở rộng khác phù hợp với các lớp bài toán trong thực tiễn.

Luận văn xin cảm ơn sự hỗ trợ nhiệt tình từ ThS. Phạm Việt Anh cùng với phòng mô phỏng và tính toán hiệu năng cao (SHPC) thuộc Viện Công nghệ HaUI cho một số thực nghiệm của luận văn.

TÀI LIỆU THAM KHẢO

[1] Pawlak, Rough sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publisher, London, 1991.

[2] D. Dubois, H. Prade, "Rough fuzzy sets and fuzzy rough sets", International Journal of General Systems 17, pp.191-209, 1990.

[3] D. Dubois, H. Prade, "Putting rough sets and fuzzy sets together", Intelligent Decision Support, Kluwer Academic Publishers, Dordrecht, 1992.

[4] Nguyen S. Hoa, Nguyen H. Son, "Some efficient alogrithms for rough set methods", Proceedings of the sixth International Conference on Information Processing Management of Uncertainty in Knowledge Based Systems, pp. 1451 – 1456, 1996.

[5] Xu Z.Y., Liu Z.P., Yang B.R. and Song W., "A quick attribute reduction algorithm with complexity of $Max\left\{O(|C||U|), O\left(|C|^2|U/C|\right)\right\}$," Journal of Computers, Vol.29, No.3, pp. 391-399, 2006.

[6] W. Shu, W. Qian and Y. Xie, "Incremental approaches for feature selection from dynamic data with the variation of multiple objects," Knowledge-Based Systems, vol. 163, pp. 320–331, 2019.

[7] Y.T. Xu, L.S. Wang, R.Y. Zhang, "A dynamic attribute reduction algorithm based on 0-1 integer programming," Knowledge-Based Systems, 24, 1341-1347, 2011.

[8] M. Yang, An incremental updating algorithm for attribute reduction based on improved discernibility matrix, Chinese Journal of Computers 30(5), 815- 822, 2007.

[9] N. L. Giang, L. H. Son, T. T. Ngan, T. M. Tuan, H. T. Phuong et al., "Novel incremental algorithms for attribute reduction from dynamic decision systems using hybrid filter-wrapper with fuzzy partition distance," IEEE Transactions on Fuzzy Systems, vol. 28, no. 5, pp. 858–873, 2020.

[10] A. K. Tiwari, S. Shreevastava, T. Som, and K. K. Shukla, "Tolerance-based intuitionistic fuzzy-rough set approach for attribute reduction," Expert Syst. With Appl., vol. 101, pp. 205–212, 2018.

[11] C. Z. Wang et al., "A fitting model for feature selection with Fuzzy rough sets," IEEE Trans. Fuzzy Syst., vol. 25, no. 4, pp. 741–753, Aug. 2017.

[12]R. Jensen and Q. Shen, "Semantics-preserving dimensionality reduction: Rough and fuzzy-rough-based approaches," IEEE Trans. Knowl. Data Eng., vol. 16, no. 12, pp. 1457–1471, 2004.

[13] R. Jensen and Q. Shen, "Fuzzy-rough sets assisted attribute reduction," IEEE Trans. Fuzzy Syst. vol. 15, no. 1, pp. 73–89, 2007.

[14] R. Jensen and Q. Shen, "New approaches to fuzzy-rough feature selection," IEEE Trans. Fuzzy Syst., vol. 17, no. 4, pp. 824–838, 2009.

[15] X. Zhang, C. L. Mei, D. G. Chen, and Y. Y. Yang, "A fuzzy rough set-based feature selection method using representative instances," Knowl.-Based Syst., vol. 151, pp. 216–229, 2018.

[16] T. K. Sheeja and A. S. Kuriakose, "A novel feature selection method using fuzzy rough sets," Comput. Ind., vol. 97, pp. 111–116, 2018.

[17] Y. H. Qian, Q. Wang, H. H. Cheng, J. Y. Liang, and C. Y. Dang, "Fuzzy-rough feature selection accelerator," Fuzzy Sets Syst., vol. 258, pp. 61–78, 2015.

[18] Y. Lin, Y. Li, C. Wang, and J. Chen, "Attribute reduction for multi-label learning with fuzzy rough set," Knowl.-Based Syst., vol. 152, pp. 51–61, 2018.

[19] D. G. Chen, L. Zhang, S. Y. Zhao, Q. H. Hu, and P. F. Zhu, "A novel algorithm for finding reducts with fuzzy rough sets," IEEE Trans. Fuzzy Syst., vol. 20, no. 2, pp. 385–389, 2012.

[20] E. C. C. Tsang, D. G. Chen, D. S. Yeung, X. Z. Wang, and J. W. T. Lee, "At- tributes reduction using fuzzy rough sets," IEEE Trans. Fuzzy Syst., vol. 16, no. 5, pp. 1130–1141, 2008.

[21] J. H. Dai, Y. J. Yan, Z. W. Li, and B. S. Liao, "Dominance-based fuzzy rough set approach for incomplete interval-valued data," J. Intell. Fuzzy Syst., vol. 34, pp. 423–436, 2018.

[22] Q. H. Hu, D. R. Yu, Z. X. Xie, and J. F. Liu, "Fuzzy probabilistic approximation spaces and their information measures," IEEE Trans. Fuzzy Syst., vol. 14, no. 2,pp. 191–201, 2006.

[23] Q. H. Hu, D. R. Yu, and Z. X. Xie, "Information-preserving hybrid data reduction based on fuzzy-rough techniques," Pattern Recognit. Lett., vol. 27, no. 5, pp. 414–423,

2016.

[24] Q. H. Hu, Z. Xie, and D. R. Yu, "Comments on fuzzy probabilistic approximations spaces and their information measures," IEEE Trans. Fuzzy Syst., vol. 16, no. 2, pp. 549–551, 2008.

[25] C. Z. Wang, Y. Huang, M. W. Shao, and X. D. Fan, "Fuzzy rough set-based attribute reduction using distance measures," Knowl.-Based Syst., vol. 164, pp. 205–212, 2019.

[26] C. Z. Wang, Y. Qi, and Q. He, "Attribute reduction using distance-based fuzzy rough sets," in Proc. Int. Conf. Mach. Learn. Cybern., pp. 860–865, 2015.

[27] C. C. Nghia, D. Janos, N. L. Giang, and V. D. Thi, "About a fuzzy distance be- tween two fuzzy partitions and attribute reduction problem," Cybern.Inf. Tech- nol., vol. 16, no. 4, pp. 13–28, 2016.

[28] Y. H. Qian., J. Y. Liang, W.-Z. Wu, and C. Y. Dang, "Information granularity in Fuzzy binary GrC model," IEEE Trans. Fuzzy Syst., vol. 19, no. 2, pp. 253–264, 2011.

[29] J. H. Dai and Q. Xu, "Attribute selection based on information gain ratio in fuzzy rough set theory with application to tumor classification," Appl.Soft Comput., vol. 13, pp. 211–221, 2013.

BM.02.03.SĐH-13

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BẢN GIẢI TRÌNH CHỈNH SỬA ĐỀ ÁN TỐT NGHIỆP

Tên đề tài đề án tốt nghiệp: Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α
Ngành đào tạo: Hệ thống thông tin
Mã ngành: 8480104
Họ và tên học viên: Trần Phi Lực
Họ và tên người hướng dẫn: Đặng Trọng Hợp
Học hàm, học vị: TS
Đơn vị công tác: Khoa Công nghệ thông tin – Trường Đại học Công nghiệp Hà Nội

NỘI DUNG GIẢI TRÌNH

TT	Ý kiến của Hội đồng	Ý kiến của học viên và nội dung chỉnh sửa
1	Chỉnh sửa về trình bày (chính tả, tài liệu tham khảo)	Học viên đã rà soát, chỉnh sửa các lỗi chính tả, bổ sung các tài liệu tham khảo theo ý kiến của Hội đồng
2	Phân tích làm rõ chương kết quả, bám sát mục tiêu đề án	Học viên tiếp thu ý kiến của Hội đồng. Đã phân tích và chỉnh sửa nội dung chương kết quả nhằm bám sát tên đề án
3	Bổ sung, giải thích các bảng biểu	Học viên tiếp thu ý kiến của Hội đồng. Đã bổ sung bảng biểu 3.1: Các bộ dữ liệu sử dụng trong thử nghiệm và giải thích các ký hiệu sử dụng trong bảng biểu

Hà Nội, ngày 5 tháng 6 năm 2024

Xác nhận của đại diện hội đồng chấm đề án tốt nghiệp

Thư ký hội đồng xác nhận! Lê Thị Anh

Ý kiến của người hướng dẫn

Đặng Trọng Hợp

Học viên ký tên

Trần Phi Lực

BM.02.03.SĐH-09

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BẢN NHẬN XÉT ĐỀ ÁN TỐT NGHIỆP

(Dùng cho người phản biện)

Tên đề tài đề án tốt nghiệp: Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α

Ngành đào tạo: Hệ thống thông tin Mã ngành: 8480104

Họ và tên học viên: Trần Phi Lực

Họ và tên người phản biện: Hoàng Văn Thông

Học hàm, học vị: TS

Nơi công tác: Trường Đại học Giao thông vận tải

Số điện thoại liên hệ:

Nhiệm vụ trong hội đồng: Phản biện 1

NỘI DUNG NHẬN XÉT

1. Tính cấp thiết của đề tài

   Để giảm thời gian tình toán và tăng chính xác của các mô hình được xây dựng bằng các thuật toán học máy thì một trong những giải pháp quan trọng là lựa chọn thuộc tính khi tập dữ liệu đầu vào có số chiều lớn, chưa nhiều thuộc tính dư thừa, không cần thiết. Việc nghiên cứu phát triển các thuật toán rút gọn các thuộc tính của tập dữ liệu để được tập dữ liệu thu gọn khi mang vào huấn luyện các mô hình học máy vẫn đảm bảo độ chính xác như sử dụng liệu gốc đang được quan tâm nghiên cứu trong nwhnxg năm gần đây. Đề tài đề án là cấp thiết và có ý nghĩa ứng dụng cao.

2. Nội dung và kết quả nghiên cứu đạt được, những đóng góp mới của đề án tốt nghiệp

   Đề án đã trình bày được một số kết quả nghiên cứu bao gồm: lý thuyết tập thô, tập thô mờ, một số phương pháp rút gọn thuộc tính cảu bảng quyết định dựa trên lý thuyết tập thô, tập hô mờ; lý thuyết tập mờ mực α và một số thuật toán rút gọn thuộc tính trên bảng quyết định cố định và bảng quyết động động như thuật toán Filter tìm tập rút gọn sử dụng khoảng cách mờ, thuật toán Filter tìm tập rút gọn sử dụng khoảng cách mờ trên bảng quyết định động,... Tiến hành thực nghiệm để đánh giá các thuật toán khác nhau trên tập dữ liệu.

Ghi chú: *Bản nhận xét yêu cầu đánh máy từ 1-2 trang A4 và gửi cho Trung tâm Đào tạo Sau đại học trước thời điểm học viên xếp lịch bảo vệ ít nhất 2 ngày.*

Đề án không có đóng góp mới về mặt lý thuyết, đóng góp chính của đề án là tập hợp các thuật toán rút gọn thuộc tính trên bảng quyết định dựa trên lý thuyết tập thô và tập thô mờ.

3. Những vấn đề hạn chế cần trao đổi

Tác giả cần xem lại toàn bộ đề án về việc tham chiếu đến các tài liệu tham khảo đặc biết là các định nghĩa, mệnh đề, định lý, thuật toán như viết trong đề án hiện nay người đọc hiểu là những nội dung của tác giả đề xuất.

Trong mục 2.1 tác giả cần xem lại câu "đề án sẽ đề xuất hai thuật toán ..." viết như thế người đọc hiểu nhầm là các thuật toán trình bày trong chương này là thuật toán mới do tác giả đề xuất.

Chương 3 cần trình bày rõ bảng quyết định mà đề án sử dụng đê thử nghiệm các thuật toán, hiện nay chưa rõ thử nghiệm trên bảng quyết định nào.

Câu hỏi trao đổi

1. Tác giả làm rõ các thuật toán trong chương 2 là những thuật toán do tác giả phát triển hay là tham khảo?

2. Giải thích múc đích của việc đưa ra biểu đồ 2.2 độ chính xác phân lớp trong khi mục tiêu là rút gọn thuộc tính

4. Sự phù hợp của tên đề tài với chương trình đào tạo, với nội dung đề án tốt nghiệp và sự trùng lặp với các công trình đã công bố; Nội dung đề án tốt nghiệp so với đề cương đã được phê duyệt; Tính hợp lý trong kết cấu đề án tốt nghiệp

Nôi dung của đề án phù hợp với tên của đề tài, phù hợp với chương trình đào tạo, theo hiểu biết của tôi thi đề án không trùng với các công trình đã công bố. Đề án có kết cấu hợp lý.

KẾT LUẬN

Tôi đồng ý để tác giả Trần Phi Lực được bảo vệ đề án tốt nghiệp trước Hội đồng đánh giá đề án tốt nghiệp.

Hà Nội, Ngày 22 tháng 05 năm 2024

Người nhận xét

(*Ký, ghi họ và tên*)

Hoàng Văn Thông

Ghi chú: *Bản nhận xét yêu cầu đánh máy từ 1-2 trang A4 và gửi cho Trung tâm Đào tạo Sau đại học trước thời điểm học viên xếp lịch bảo vệ ít nhất 2 ngày.*

BM.02.03.SĐH-09

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BẢN NHẬN XÉT ĐỀ ÁN TỐT NGHIỆP

(Dùng cho người phản biện)

Tên đề tài đề án tốt nghiệp: Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α

Ngành đào tạo: Hệ thống thông tin Mã ngành: 8480104

Họ và tên học viên: Trần Phi Lực

Họ và tên người phản biện: Kim Đình Thái

Học hàm, học vị: Tiến sĩ

Nơi công tác: Trường Quốc tế, Đại học Quốc Gia Hà Nội

Số điện thoại liên hệ: 0966.575.484

Nhiệm vụ trong hội đồng: Phản biện 2

NỘI DUNG NHẬN XÉT

1. Tính cấp thiết của đề tài

   Đề tài này đóng vai trò quan trọng trong lĩnh vực khai thác dữ liệu và phân tích thông tin. Sự cần thiết của đề tài này phát xuất từ thực tế là các bảng quyết định trong thời đại số thường xuyên biến đổi và mở rộng, đòi hỏi các phương pháp phân tích phải linh hoạt và hiệu quả để xử lý dữ liệu lớn và động. Nghiên cứu này hướng tới việc phát triển các thuật toán mới có khả năng cải thiện độ chính xác và giảm độ phức tạp của các mô hình, đồng thời đáp ứng nhu cầu thực tiễn trong việc lựa chọn thuộc tính hiệu quả, nhất là trong các ứng dụng yêu cầu cao về tốc độ và độ chính xác.

2. Nội dung và kết quả nghiên cứu đạt được, những đóng góp mới của đề án tốt nghiệp

   Đề tài đã thành công trong việc đề xuất các thuật toán gia tăng mới, giúp cải thiện độ chính xác và giảm độ phức tạp trong quá trình lựa chọn thuộc tính, đặc biệt quan trọng đối với các bảng quyết định động. Đóng góp nổi bật của đề tài này là việc tích hợp lý thuyết tập mờ với thuật toán gia tăng, mang lại một hướng tiếp cận mới mẻ và hiệu quả, phù hợp với yêu cầu về tốc độ và độ chính xác cao trong các ứng dụng thực tiễn. Những kết quả thực nghiệm được trình bày đã chứng minh tính ưu việt của các phương pháp được đề xuất so với các nghiên cứu trước đây, đóng góp quan trọng vào lĩnh vực khai thác dữ liệu và phân tích thông tin.

3. Những vấn đề hạn chế cần trao đổi

Ghi chú: *Bản nhận xét yêu cầu đánh máy từ 1-2 trang A4 và gửi cho Trung tâm Đào tạo Sau đại học trước thời điểm học viên xếp lịch bảo vệ ít nhất 2 ngày.*

Đầu tiên, các phương pháp đề xuất chỉ tập trung vào các bảng quyết định có giá trị liên tục mà chưa xử lý tốt các bảng quyết định có nhiễu và không nhất quán, điều này có thể ảnh hưởng đến tính ứng dụng thực tế trong môi trường dữ liệu phức tạp. Thứ hai, nghiên cứu chủ yếu tập trung vào lý thuyết và thực nghiệm trên các bộ dữ liệu tiêu chuẩn, chưa thử nghiệm rộng rãi trên các bộ dữ liệu thực tế lớn với độ phức tạp cao hơn. Điều này có thể hạn chế khả năng tổng quát hóa và ứng dụng của các thuật toán trong các tình huống thực tế đa dạng. Hơn nữa, chưa có mô tả chi tiết về các bộ dữ liệu sử dụng và không tìm thấy trong tài liệu tham khảo. Cuối cùng, các thuật toán gia tăng rút gọn thuộc tính mặc dù có cải thiện đáng kể nhưng vẫn cần tối ưu hóa thêm để giảm thời gian xử lý và nâng cao hiệu suất trên các hệ thống thông tin lớn.

4. Sự phù hợp cua tên đề tài với chương trình đào tạo, với nội dung đề án tốt nghiệp và sự trùng lặp với các công trình đã công bố; Nội dung đề án tốt nghiệp so với đề cương đã được phê duyệt; Tính hợp lý trong kết cấu đề án tốt nghiệp.

   Nội dung của đề tài là phù hợp với chương trình đào tạo, không trùng lặp với công trình đã công bố, đáp ứng mục tiêu của đề cương đã được phê duyệt và có kết cấu hợp lý.

KẾT LUẬN

Tôi đồng ý (không đồng ý) để tác giả Trần Phi Lực được bảo vệ đề án tốt nghiệp trước Hội đồng đánh giá đề án tốt nghiệp.

Hà Nội, Ngày 24 tháng 05 năm 2024

Người nhận xét

(Ký, ghi họ và tên)

Kim Đình Thái

Ghi chú: *Ban nhận xét yêu cầu đánh máy từ 1-2 trang A4 và gửi cho Trung tâm Đào tạo Sau đại học trước thời điểm học viên xếp lịch bảo vệ ít nhất 2 ngày.*

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

BIÊN BẢN
HỌP HỘI ĐỒNG ĐÁNH GIÁ ĐỀ ÁN TỐT NGHIỆP

Căn cứ Quyết định số 614/QĐ-ĐHCN ngày 17/5/2024 của Hiệu trưởng trường Đại học Công nghiệp Hà Nội về việc thành lập Hội đồng đánh giá đề án tốt nghiệp, Hội đồng đã họp vào hồi...9..giờ..10..phút, ngày 24/5/2024, tại phòng.....402....nhà A1, Trường ĐH Công nghiệp HN để đánh giá luận đề án tốt nghiệp cho học viên: **Trần Phi Lực;** mã học viên: 2022700047; ngành: HTTT; mã số: 8480104.

Tên đề tài: Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lắt cắt a.

Người hướng dẫn: TS. Đặng Trọng Hợp - Trường ĐH Công nghiệp Hà Nội.

Các thành viên của Hội đồng đánh giá đề án tốt nghiệp có mặt: ..05.../...05...

Stt	Họ và tên, học hàm/học vị	Cơ quan công tác	Nhiệm vụ
1	TS. Phạm Văn Hà	Trường ĐH Công nghiệp Hà Nội	Chủ tịch
2	TS. Lê Thị Anh	Trường ĐH Công nghiệp Hà Nội	Thư ký
3	TS. Hoàng Văn Thông	Trường ĐH Giao thông Vận Tải	Phản biện 1
4	TS. Kim Đình Thái	Đại học Quốc Gia Hà Nội	Phản biện 2
6	TS. Hà Mạnh Đào	Trường ĐH Công nghiệp Hà Nội	Ủy viên

Các nội dung thực hiện:

1- Chủ tịch Hội đồng điều khiển buổi họp. Công bố Quyết định của Trường Đại học Công nghiệp Hà Nội về việc thành lập Hội đồng đánh giá đề án tốt nghiệp.

2- Thư ký Hội đồng đọc lý lịch khoa học và các điều kiện bảo vệ đề án tốt nghiệp của học viên (Có bản lý lịch khoa học và kết quả các học phần của học viên kèm theo).

3- Học viên trình bày tóm tắt đề án tốt nghiệp.

4- Phản biện đọc nhận xét (có văn bản kèm theo)

5- Các câu hỏi của thành viên Hội đồng:

(1) Tại sao lại đề xuất thuật toán trong chương 2 là do tác giả đề xuất hay tham khảo? (2) Mục đích của việc đưa ra biểu đồ đặc trưng chất lượng của phân lớp, mang ý nghĩa gì? (3) Có sử dụng lắt cắt α thì lắt cắt α được lựa chọn như thế nào? Có ảnh hưởng đến kết quả hay không? (4) Các tập dữ liệu sử dụng như thế nào? Tham số đánh giá?
(5) Ngoài lắt cắt α còn cách tiếp cận nào không?

6. Trả lời của học viên:

Học viên trả lời đầy đủ các câu hỏi của hội đồng. Hội đồng đánh giá học viên hiểu tương đối tốt câu hỏi và trả lời tương đối rõ ràng, chính xác, tuy nhiên vẫn còn một số vấn đề cần trao đổi sâu hơn.

7. Người hướng dẫn hoặc thư ký đọc nhận xét về quá trình thực hiện đề án tốt nghiệp của học viên *(có văn bản kèm theo).*

Cán bộ hướng dẫn đồng ý cho học viên được bảo vệ tốt nghiệp và đánh giá thái độ học viên tích cực chủ động trong quá trình thực hiện đề án.

8. Ý kiến của Hội đồng (Cần nêu rõ các vấn đề của đề án tốt nghiệp cần chỉnh sửa (nếu có)):

Hội đồng đồng ý đánh giá đề án đạt yêu cầu với các chỉnh sửa:

Chỉnh sửa về trình bày (chính tả, tài liệu tham khảo...)

Phân tích làm rõ chương kết quả khảo sát mục tiêu đề án.

Bổ sung giải thích các bảng biểu...

Yêu cầu xác nhận của Hội đồng sau khi chỉnh sửa: Có: ☑ Không: ☐

Thành viên xác nhận: TS. Lê Thị Anh. Nhiệm vụ trong HĐ: Thư ký

9. Hội đồng họp và tổng hợp kết quả

- Chủ tịch HĐ công bố điểm đánh giá đề án tốt nghiệp của từng thành viên và điểm trung bình đề án tốt nghiệp *(có Phiếu đánh giá đề án tốt nghiệp và Phiếu tổng hợp điểm đánh giá đề án tốt nghiệp kèm theo);*

- **Điểm đề án tốt nghiệp:** ***Bằng số*** 8.6 ***Bằng chữ***: Tám phẩy sáu

Kết luận: Đề án tốt nghiệp (đạt hoặc không đạt) Đạt yêu cầu là một đề án tốt nghiệp thạc sĩ theo Quy chế đào tạo trình độ thạc sĩ hiện hành./.

CHỦ TỊCH HỘI ĐỒNG	**THƯ KÝ HỘI ĐỒNG**
Phạm Văn Hà	Lê Thị Anh

Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α

by Lực Trần Phi

Submission date: 03-May-2024 11:59PM (UTC+0700)
Submission ID: 2370040916
File name: 1801_Luc_Tran_Phi_Nghien_cuu_mot_so_thuat_toan_gia_tang_lua_chon_thuoc_tinh_tren_bang_quyet__inh__ong_theo_tie_863308970.docx (654.65K)
Word count: 13605
Character count: 55189

Nghiên cứu một số thuật toán gia tăng lựa chọn thuộc tính trên bảng quyết định động theo tiếp cận tập mờ sử dụng lát cắt α

ORIGINALITY REPORT

20%	17%	16%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Nguyễn Long Giang, Phạm Minh Ngọc Hà, Nguyễn Văn Thiện, Nguyễn Bá Quảng. "Về một thuật toán gia tăng tìm tập rút gọn của bảng quyết định không đầy đủ", Research and Development on Information and Communication Technology, 2019 Publication	4%
2	ioit.ac.vn Internet Source	3%
3	www.thuvientailieu.vn Internet Source	2%
4	jst.tnu.edu.vn Internet Source	2%
5	vjs.ac.vn Internet Source	2%
6	tailieudientu.lrc.tnu.edu.vn Internet Source	2%

7	Tran Thanh Dai, Nguyen Long Giang, Hoang Thi Minh Chau, Tran Thi Ngan. "APPROACH FOR ATTRIBUTE SUBSET SELECTION BASED INTUITIONISTIC FUZZY-ROUGH SET", KỶ YẾU HỘI NGHỊ KHOA HỌC CÔNG NGHỆ QUỐC GIA LẦN THỨ XIII NGHIÊN CỨU CƠ BẢN VÀ ỨNG DỤNG CÔNG NGHỆ THÔNG TIN - Proceedings of the 13th National Conference on Fundamental & Applied Information Technology Research, 2020 Publication	1 %
8	Nguyễn Long Giang, Nguyễn Văn Thiện, Cao Chính Nghĩa. "VỀ PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRỰC TIẾP TRÊN BẢNG QUYẾT ĐỊNH SỬ DỤNG KHOẢNG CÁCH MỜ", FAIR - NGHIÊN CỨU CƠ BẢN VÀ ỨNG DỤNG CÔNG NGHỆ THÔNG TIN - 2016, 2017 Publication	1 %
9	toc.123doc.org Internet Source	1 %
10	Vũ Văn Định, Vũ Đức Thi, Nguyễn Long Giang, Ngô Quốc Tạo. "Phương pháp rút gọn thuộc tính trong bảng quyết định không đầy đủ sử dụng khoảng cách phân hoạch", Các công trình nghiên cứu, phát triển và ứng dụng Công nghệ Thông tin và Truyền thông, 2015 Publication	1 %

Hanoi Pedagogycal University 2

11 Publication 1%

Exclude quotes On
Exclude bibliography On
Exclude matches < 100 words