BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Trần Thanh Đại

RÚT GỌN THUỘC TÍNH TRONG BẢNG QUYẾT ĐỊNH THEO TIẾP CẬN TẬP THÔ MỜ TRỰC CẢM VÀ TÔPÔ SUY RỘNG

LUẬN ÁN TIẾN SĨ NGÀNH HỆ THỐNG THÔNG TIN

Hà Nội - Năm 2023

BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Trần Thanh Đại

RÚT GỌN THUỘC TÍNH TRONG BẢNG QUYẾT ĐỊNH THEO TIẾP CẬN TẬP THÔ MỜ TRỰC CẢM VÀ TÔPÔ SUY RỘNG

LUẬN ÁN TIẾN SĨ NGÀNH HỆ THỐNG THÔNG TIN
Mã số: 9 48 01 04

Xác nhận của Học viện Khoa học và Công nghệ	Người hướng dẫn 1 *(Ký, ghi rõ họ tên)*	Người hướng dẫn 2 *(Ký, ghi rõ họ tên)*
KT. GIÁM ĐỐC PHÓ GIÁM ĐỐC Nguyễn Thị Trung	Nguyễn Long Giang	Vũ Đức Thi

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Hà Nội - Năm 2023

LỜI CAM ĐOAN

Tôi xin được cam đoan đây là công trình nghiên cứu của riêng tôi dưới sự hướng dẫn của PGS. TS Nguyễn Long Giang và GS. TS Vũ Đức Thi tại Viện Công nghệ thông tin - Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Các kết quả nghiên cứu lý thuyết và thực nghiệm trong luận án này được trình bày chính xác, trung thực và không sao chép từ bất kỳ nguồn tài liệu nào và dưới bất kỳ hình thức nào. Việc tham khảo các nguồn tại liệu được trích dẫn và ghi nguồn đầy đủ.

Hà Nội, ngày 20 tháng 11 năm 2023

Trần Thanh Đại

LỜI CẢM ƠN

Luận án này được hoàn thiện với sự nỗ lực và cố gắng không ngừng của tác giả cùng với sự ân cần chỉ bảo, giúp đỡ của các thầy hướng dẫn, sự góp ý xác đáng của các chuyên gia, nhà khoa học, sự động viên về tinh thần của gia đình, bạn bè và đồng nghiệp trong suốt quá trình học tập và nghiên cứu của tác giả.

Trước tiên, tác giả xin bày tỏ lòng biết ơn đến PGS. TS Nguyễn Long Giang, GS. TS Vũ Đức Thi đã tận tình chỉ bảo, hướng dẫn và động viên tác giả hoàn thành luận án đúng mục tiêu và đúng tiến độ. Tác giả cũng xin được bày tỏ lời cảm ơn sâu sắc đến PGS. TS Lê Hoàng Sơn đã có những góp ý quý giá trong suốt quá trình thực hiện luận án này.

Tác giả xin gửi lời cảm ơn tới các thầy, cô giáo và cán bộ Phòng Đào tạo tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Đặc biệt tác giả xin trân trọng cảm ơn Phòng Quản lý sau Đại học của Viện Công nghệ thông tin đã nhiệt tình giúp đỡ, tạo ra môi trường nghiên cứu thuận lợi cho tác giả hoàn thành luận án đúng tiến độ và đúng quy định của Học viện.

Tác giả xin chân thành cảm ơn nhóm nghiên cứu AI 4.0 tại Viện Công nghệ thông tin (ITI) - Đại học Quốc gia Hà Nội đã giúp đỡ tác giả về mặt chuyên môn và tinh thần nghiên cứu trong suốt quá trình trao đổi và nghiên cứu học thuật tại ITI.

Tác giả xin chân thành cảm ơn tới Ban Giám Hiệu Trường Đại học Kinh tế Kỹ thuật Công nghiệp đã động viên tinh thần và tạo nhiều điều kiện thuận lợi trong suốt quá trình học tập và nghiên cứu.

Đặc biệt tác giả xin bày tỏ lòng biết ơn sâu sắc tới Gia đình và người thân đã hi sinh vô điều kiện, tạo điều kiện tốt nhất về tinh thần và thời gian cho tác giả trong suốt quá trình học tập và làm nghiên cứu.

NCS Trần Thanh Đại

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ, CÁC CHỮ VIẾT TẮT

ACC	Độ chính xác (Accuracy)
CLS	Miền đóng
IS	Hệ thông tin (Information System)
DT	Bảng quyết định (Decision Table)
FN	False Negative (Phủ định sai)
FP	False Positive (Khẳng định sai)
TN	True Negative (Phủ định đúng)
TP	True Positive (Khẳng định đúng)
Base	Cơ sở
Subbase	Cơ sở con
IF-base	Cơ sở mờ trực cảm (Intuittionistic Fuzzy Base)
IF-subbase	Cơ sở con mờ trực cảm (Intuittionistic Fuzzy Subbase)
FRS	Tập thô mờ (Fuzzy Rough Set)
IFRS	Tập thô mờ trực cảm (Intuitionistic Fuzzy Rough Set)
IFS	Tập mờ trực cảm (Intuitionistic Fuzzy Set)
IFT	Tôpô mờ trực cảm (Intuitionistic Fuzzy Topology)
INT	Miền trong
NRS	Tập thô lân cận (Neighborhood Rough set)
PRS	Tập thô xác suất (Probality Rough set)
VPRS	Tập thô điều chỉnh chính xác (Variable Precision Rough Set)
IFIE	Entropy thông tin mờ trực cảm
IFD	Khoảng cách mờ trực cảm (Intuitionistic Fuzzy Distance)
k-NN	k- láng giềng gần nhất (k - Nearest Neighbor)
SVM	Máy vector hỗ trợ (Supper Vector Meachine)

DANH MỤC CÁC KÝ HIỆU

C	Tập thuộc tính điều kiện
D	Tập thuộc tính quyết định
U	Tập đối tượng
O	Big-O
$\mathbb{R}$	Tập số thực
$\mathbb{T}$	Thời gian thực hiện của mô hình phân lớp
M	Ma trận quan hệ
M^T	Ma trận chuyển vị
$M^{\geq}$	Ma trận quan hệ ưu tiên
Model	Mô hình phân lớp
R	Quan hệ tương đương
$R^{\geq}$	Quan hệ ưu tiên
R^{β}	Quan hệ ưu tiên mờ ngưỡng β
W_A	wrapper thuộc tính (wrapper attribute)
W_δ	wrapper theo giá trị delta
RAW	Dữ liệu ban đầu
|C|	Số lượng các thuộc tính điều kiện trong tập C
|U|	Số lượng các đối tượng trong tập U
$\overset{\delta}{=}$	delta - equal
$\overset{\approx}{d}$	Khoảng cách mờ trực cảm
$[\overset{\approx}{X}]$	Phân hoạch mờ trực cảm của tập thuộc tính X
ζ	Độ đo tương đồng giữa hai tôpô mờ trực cảm
$\mathscr{T}$	Tôpô theo tiếp cận tập thô
$\prec$	Quan hệ thứ tự bộ phận

DANH MỤC CÁC HÌNH VẼ

DANH MỤC CÁC BẢNG BIỂU

MỞ ĐẦU

Tính cấp thiết của đề tài luận án

Rút gọn thuộc tính [1, 2, 3] hay chọn lọc thuộc tính là bước tiền xử lý dữ liệu quan trọng, được ứng dụng rộng rãi trong các lĩnh vực liên quan đến nhận dạng mẫu và khai thác dữ liệu như phân lớp dữ liệu [4, 5], nhận dạng chữ viết tay [6, 7], nhận dạng tiếng nói [8, 9], phát hiện và phân loại spam [10, 11] và hỗ trợ ra quyết định [12, 13]. Khác với các phương pháp giảm chiều theo tiếp cận học sâu, tập rút gọn thu được vẫn giữ được nguyên thông tin của các giá trị thuộc tính ban đầu, là cơ sở để hiểu được dữ liệu trong quá trình ra quyết định. Do đó, rút gọn thuộc tính là bài toán đi tìm tập con của tập thuộc tính ban đầu có liên quan nhiều nhất hoặc loại bỏ các thuộc tính dư thừa, ít liên quan trong bảng quyết định. Rút gọn thuộc tính thường được thực hiện để mô hình đạt được một số mục tiêu như tăng tính dễ hiểu của luật, cải thiện hiệu năng phân lớp, giảm chi phí xây dựng mô hình.

Mô hình lý thuyết tập thô (Rough Set - RS) được Pawlack giới thiệu vào năm 1982 là công cụ toán học mạnh mẽ, ứng dụng hiệu quả cho các trường hợp dữ liệu không chắc chắn, không đầy đủ và thiếu nhất quán [14]. Rút gọn thuộc tính là một trong những ứng dụng nổi bật của RS, đã và đang nhận được sự quan tâm của các nhà nghiên cứu [15, 16, 17]. Bên cạnh đó, không gian tôpô theo tiếp cận RS [18, 19] cũng là một khái niệm quan trọng được Pawlack và các cộng sự giới thiệu vào năm 1988, nhận được nhiều sự quan tâm của nhiều nhà nghiên cứu [4, 20, 21] trong những năm gần đây.

Hơn ba thập kỉ vừa qua, hướng rút gọn thuộc tính theo tiếp cận RS đã và đang thu hút được sự quan tâm của nhiều nhà nghiên cứu. Các kết quả nghiên cứu cho thấy phương pháp rút gọn thuộc tính theo tiếp cận RS hiệu quả trên các bảng quyết định có thuộc tính giá trị rời rạc. Tuy nhiên, với các bảng quyết định có thuộc tính giá trị liên tục (bảng quyết định số) cần phải thực hiện bước biến đổi miền giá trị liên tục về

miền giá trị rời rạc trước khi sử dụng RS để rút gọn thuộc tính. Bước biến đổi này làm tăng chi phí tính toán và có thể làm mất dữ liệu trong quá trình biến đổi. Do đó, các nhà nghiên cứu đề xuất mở rộng RS trên nền tập mờ (Fuzzy Set - FS), ứng dụng xây dựng phương pháp rút gọn thuộc tính trực tiếp trên các bảng quyết định số.

Mô hình tập thô mờ (Fuzzy Rough Set - FRS) [22] là sự kết hợp của RS và FS. FRS sử dụng quan hệ tương tự thay cho quan hệ tương đương để xây dựng không gian xấp xỉ. Do đó, quan hệ của các đối tượng trong một tập được biểu diễn trơn hơn so với quan hệ tương đương truyền thống. Cho đến nay, các hướng nghiên cứu rút gọn thuộc tính theo tiếp cận FRS diễn ra khá sôi động với các độ đo mới được đề xuất như: độ đo miền dương mờ (Fuzzy POS - FPOS) [23, 24, 17, 25, 26, 27, 28, 29], độ đo entropy thông tin mờ (Fuzzy Information Entropy - FIE) [30, 13, 31, 32], độ đo khoảng cách mờ (Fuzzy Distance - FD) [33]. Tuy nhiên, các bảng quyết định số thường chứa các thuộc tính có độ nhất quán thấp (nhiễu). Do đó, các kết quả thực nghiệm của các nghiên cứu cho thấy tập rút gọn thu được còn hạn chế về độ chính xác phân lớp trên các tập dữ liệu nhiễu. Gần đây các nhà nghiên cứu đề xuất mở rộng RS trên nền tập mờ trực cảm (Intuitionistic Fuzzy Set - IFS) nhằm khắc phục nhược điểm trên của FRS.

Mô hình tập thô mờ trực cảm (Intuitinistic Fuzzy Rough Set - IFRS) là sự kết hợp của RS và IFS. Khác với mô hình FRS, mỗi phần tử trong IFRS đặc trưng bởi hai thành phần, độ thành viên và độ không thành viên. Do đó, khả năng xấp xỉ một đối tượng vào một tập mục tiêu được cho là chính xác hơn so với tiếp cận FRS. Hơn nữa, không gian xấp xỉ trên tập nền IFS biểu diễn mối quan hệ của hai đối tượng được cho là chặt hơn so với tập nền FS [34, 35]. Do đó, các nhà nghiên cứu nhận định phương pháp rút gọn thuộc tính theo tiếp cận IFRS có khả năng cho tập rút gọn có độ chính xác phân lớp tốt hơn trên các tập dữ liệu nhiễu. Gần đây, các công bố điển hình về rút gọn thuộc tính theo tiếp cận IFRS gồm có: phương pháp rút gọn thuộc tính theo tiếp cận miền dương mờ trực cảm (Intuitionistic Fuzzy POS) [36], phương pháp rút gọn thuộc tính theo tiếp cận entropy thông tin mờ trực cảm (Intuitionistic Fuzzy

Information Entropy - IFIE) [15]. Tuy nhiên, các kết quả thực nghiệm của các nghiên cứu này cho thấy, độ chính xác phân lớp của tập rút gọn thu được trên các bộ dữ liệu nhiễu chưa được cải thiện đáng kể.

Năm 1987, Pawlack và các cộng sự cũng chỉ ra mối quan hệ về cấu trúc của tập thô và tôpô [37], kéo theo xu hướng rút gọn thuộc tính theo tiếp cận tôpô ngày càng sôi động. Năm 2005, lần đầu tiên Lashin và các cộng sự đề xuất khái niệm tôpô rút gọn [38] dựa trên cơ sở và cơ sở con của tôpô. Từ đó, các phương pháp xây dựng tôpô theo tiếp cận tập thô ngày càng được quan tâm và phát triển [39, 4].

Để rút gọn thuộc tính cho bảng quyết định theo tiếp cận tôpô, trước tiên cần phải xây dựng được cấu trúc tôpô dựa trên các thông tin đã có trong bảng quyết định. Đây là một thách thức lớn, đã và đang thu hút được sự quan tâm của nhiều nhà nghiên cứu [38, 40, 39, 20]. Hiện nay có hai phương pháp chính để xây dựng tôpô gồm có: phương pháp xây dựng tôpô từ không gian xấp xỉ [41, 42, 43], phương pháp xây dựng tôpô theo tiếp cận tập thô [39, 44].

Bên cạnh đó, mối quan hệ của không gian xấp xỉ của tập thô và tôpô được các nhà nghiên cứu sử dụng để xác định, trong trường hợp nào các phép toán xấp xỉ của RS và tôpô là tương đương [45, 46, 44, 47, 48, 39]. Mối quan hệ của các phép toán xấp xỉ của RS và tôpô [49, 20, 46, 50] được các nhà nghiên cứu sử dụng để xác định, trong trường hợp nào không gian xấp xỉ và tôpô là tương đương [45, 51, 52]. Gần đây, Xie và các công sự [21] đã đề xuất phương pháp rút gọn thuộc tính theo tiếp cận ma trận phân biệt tôpô. Tuy nhiên tiếp cận ma trận phân biệt là phương pháp tìm tất cả các tập rút gọn có thể, hơn nữa phương pháp tiếp cận này đã được chứng minh là phương pháp tốn kém nhất về tài nguyên thời gian và không gian tính toán.

Tại Việt Nam, đã có một số luận án tiến sĩ nghiên cứu phương pháp rút gọn thuộc tính trực tiếp trên bảng quyết định số gồm có: luận án tiến sĩ của tác giả Cao Chính Nghĩa [3], nghiên cứu rút gọn thuộc tính và sinh luật quyết định trên các bảng dữ liệu số. Luận án tiến sĩ của tác giả Nguyễn Văn Thiện [2], đề xuất độ đo khoảng cách mờ và xây dựng một số thuật toán tìm tập rút gọn. Luận án tiến sĩ của tác giả Hồ Thị

Phượng [1], đề xuất một số thuật toán gia tăng tìm tập rút gọn trong các bảng quyết định động sử dụng độ đo khoảng cách mờ. Từ các kết quả khảo sát bên trên cho thấy, các phương pháp rút gọn thuộc tính trực tiếp trên bảng quyết định số tại Việt Nam hiện nay chủ yếu dựa trên tiếp cận FRS. Các kết quả thực nghiệm cho thấy tập rút gọn thu được theo tiếp cận FRS còn chưa hiệu quả về kích thước và độ chính xác phân lớp trên các bộ dữ liệu nhiễu.

Mục tiêu nghiên cứu

Xuất phát từ những vấn đề còn tồn tại của các phương pháp rút gọn thuộc tính theo tiếp cận IFRS và tôpô hiện nay, luận án đặt ra hai mục tiêu nghiên cứu chính như sau:

1) Nghiên cứu phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm: Với phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm, *vấn đề nghiên cứu trước tiên* là cần tìm hiểu cách thức mô tả mối quan hệ của các đối tượng hiệu quả trên nền tập mờ trực cảm, cụ thể là xây dựng các hàm đánh giá độ thuộc và độ không thuộc hiệu quả cho không gian xấp xỉ mờ trực cảm. Trên cơ sở đó, *vấn đề nghiên cứu tiếp theo* là cần xây dựng độ đo khoảng cách giữa các phân hoạch mờ trực cảm, làm cơ sở đề xuất độ đo đánh giá độ quan trọng của thuộc tính. Cuối cùng là ứng dụng độ đo đề xuất để xây dựng thuật toán rút gọn thuộc tính hiệu quả trên các bộ dữ liệu nhiễu.

2) Nghiên cứu phương pháp rút gọn thuộc tính theo tiếp cận tôpô đại số: Theo phương pháp tôpô đại số, *vấn đề nghiên cứu trước tiên* là cần tìm hiểu các phương pháp xây dựng cấu trúc tôpô trên không gian các thuộc tính, nghiên cứu các tính chất của tôpô sao cho có thể đánh giá tôpô trong một không gian nhỏ hơn để tiết kiệm chi phí tính toán. Trên cơ sở đó, *vấn đề nghiên cứu tiếp theo* là nghiên cứu các phép toán cơ bản trên cấu trúc tôpô thuộc tính nhằm xây dựng các phương pháp đánh giá, nhận diện độ quan trọng của thuộc tính, định nghĩa tập rút gọn thông qua cấu trúc tôpô. Cuối cùng là ứng dụng xây dựng thuật toán rút gọn thuộc tính hiệu quả trên các bộ dữ liệu nhiều, các bộ dữ liệu có số chiều lớn trong thực tiễn.

Đối tượng nghiên cứu

Luận án tập trung nghiên cứu các phương pháp rút gọn thuộc tính trong bảng quyết định theo tiếp cận tập thô và tôpô đại số gồm có:

- Khảo sát các khái niệm cơ bản về tập thô, các độ đo được sử dụng để đánh giá độ quan trọng của thuộc tính và các phương pháp xây dựng thuật toán rút gọn thuộc tính theo tiếp cận Heuristic.

- Khảo sát các khái niệm cơ bản về tôpô theo tiếp cận tập thô, tôpô thu từ không gian xấp xỉ, tôpô thu từ quan hệ của các phép toán xấp xỉ, tính khả li trong không gian tôpô và tôpô rút gọn.

Phạm vi nghiên cứu

Rút gọn thuộc tính là bài toán đi tìm tập con của tập thuộc tính ban đầu trong khi vẫn giữ được thông tin mô tả của các đối tượng trong bảng quyết định. Do đó, luận án tập trung nghiên cứu các phương pháp rút gọn thuộc tính dựa trên biến thể của tập thô và tôpô đại số như sau:

- Nghiên cứu các mô hình tập thô mở rộng trên nền tập mờ và tập mờ trực cảm, ứng dụng xây dựng thuật toán rút gọn thuộc tính trong bảng quyết định số.

- Nghiên cứu cấu trúc tôpô theo tiếp cận tập thô và không gian xấp xỉ trên nền tập mờ và tập mờ trực cảm, ứng dụng xây dựng thuật toán rút gọn thuộc tính trong bảng quyết định số.

Phương pháp nghiên cứu:

Các kết quả nghiên cứu của luận án được đánh giá trên hai góc độ nghiên cứu gồm có:

- *Góc độ nghiên cứu lý thuyết*: các định nghĩa được trình bày rõ ràng, các mệnh đề được chứng minh chặt chẽ dựa vào nền tảng cơ bản của lý thuyết tập hợp, độ đo, tập thô, tập mờ, tập mờ trực cảm và entropy Shanon.

- *Góc độ nghiên cứu thực nghiệm*: các thuật toán được cài đặt và thực nghiệm trên các bộ dữ liệu từ UCI[1]. Sử dụng các mô hình phân lớp dữ liệu phù hợp với dữ liệu và

các độ đo đánh giá, phương pháp đánh giá nhằm đánh giá chất lượng của tập rút gọn. So sánh chất lượng tập rút gọn từ thuật toán đề xuất với các thuật toán khác nhằm củng cố giả thiết nghiên cứu của luận án là hoàn toàn hợp lý.

Cấu trúc của luận án:

Ngoài phần mở đầu và kết luận, luận án có 04 chương nội dung nghiên cứu như sau:

Chương 1. Luận án giới thiệu và định nghĩa bài toán rút gọn thuộc tính, phân loại các phương pháp rút gọn thuộc tính. Trình bày các khái niệm cơ bản về hệ thông tin, bảng quyết định và tập rút gọn. Trình bày các khái niệm cơ bản về mô hình tập thô truyền thống, không gian tôpô và tập mờ trực cảm. Trên cơ sở đó, luận án trình bày các phương pháp rút gọn thuộc tính theo tiếp cận độ đo và tiếp cận tôpô. Trình bày các chỉ số và phương pháp đánh giá chất lượng mô hình phân lớp dữ liệu. Các đóng góp chính của luận án được trình bày trong các chương 2, chương 3, và chương 4.

Chương 2. Luận án trình bày phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm bao gồm các bước chính như sau:

- Mở rộng độ đo khoảng cách mờ trên nền tập mờ trực cảm và xây dựng độ đo khoảng cách giữa các phân hoạch mờ trực cảm.

- Xây dựng độ đo đánh giá độ quan trọng của thuộc tính dựa trên khoảng cách giữa các phân hoạch mờ trực cảm, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm δ-equal để định nghĩa tập rút gọn và xây dựng điều kiện dừng của thuật toán

Trên cơ sở đó, luận án đề xuất phương pháp lai ghép filter - wrapper tìm tập rút gọn hiệu quả về độ chính xác phân lớp trên các tập dữ liệu nhiễu. Ngoài ứng dụng của độ đo đề xuất cho bài toán rút gọn thuộc tính, độ đo này có thể áp dụng cho một số bài toán phân lớp, dự báo, hỗ trợ ra quyết định có liên quan đến kĩ thuật tính toán mềm trên tập các số mờ trực cảm.

[1]https://archive.ics.uci.edu/ml/datasets.html

Chương 3. Luận án trình bày phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm bao gồm các bước chính như sau:

- Đề xuất công thức quan hệ ưu tiên mờ trực cảm để xây dựng không gian xấp xỉ mờ trực cảm.

- Đề xuất cấu trúc tôpô mờ trực cảm dựa trên các cơ sở và cơ sở con được định nghĩa theo không gian xấp xỉ mờ trực cảm.

- Đề xuất độ đo tương đồng giữa các tôpô dựa trên khoảng cách tương đồng giữa các cơ sở con, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm tôpô đơn vị, làm cơ sở để định nghĩa tập rút gọn và xây dựng điều kiện dừng của thuật toán.

Trên cơ sở đó, luận án đề xuất hai phương pháp tìm tập rút gọn như sau:

- Đề xuất phương pháp filter thuộc tính cho tập rút gọn hiệu quả về thời gian và kích thước.

- Đề xuất phương pháp lai ghép filter - wrapper kết hợp cấu trúc dữ liệu ngắn xếp (Stack) cho tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp.

Chương 4. Luận án trình bày phương pháp rút gọn thuộc tính theo tiếp cận tôpô Hausdorff bao gồm các bước chính như sau:

- Xây dựng cấu trúc tôpô theo tiếp cận tập thô trên không gian xấp xỉ mờ trực cảm ngưỡng β.

- Đề xuất phương pháp xác định cấu trúc tôpô Hausdorff của thuộc tính, làm cơ sở để xây dựng phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm đồng cấu trúc thuộc tính, làm cơ sở để phân nhóm các thuộc tính có cấu trúc tôpô Hausdorff.

Trên cơ sở đó, luận án đề xuất phương pháp wrapper các nhóm thuộc tính cho tập rút gọn hiệu quả về thời gian và độ chính xác phân lớp.

Cuối cùng, phần kết luận sẽ trình bày những kết quả đã đạt được của luận án, hướng phát triển trong tương lai và những vấn đề quan tâm của tác giả.

CHƯƠNG 1. TỔNG QUAN BÀI TOÁN RÚT GỌN THUỘC TÍNH THEO TIẾP CẬN TẬP THÔ VÀ TÔPÔ

1.1. Mở đầu

Rút gọn thuộc tính (attribute reduction) hay còn được gọi lựa chọn đặc trưng (feature selection) là một trong những bước tiền xử lý dữ liệu quan trọng trong các lĩnh vực nhận dạng (pattern recognition), học máy (machine learning) và khai thác dữ liệu (data mining). Đối với các tập dữ liệu dành cho các bài toán học không giám sát (unsupervised - learning), rút gọn thuộc tính nhằm lựa chọn một tập con của tập thuộc tính ban đầu bảo toàn thông tin của tập thuộc tính gốc. Đối với các tập dữ liệu cho các bài toán học có giám sát (supervised - learning), rút gọn thuộc tính nhằm chọn ra một tập con của tập thuộc tính ban đầu bảo toàn khả năng phân lớp hay dự báo so với tập thuộc tính gốc [53].

Có ba tiếp cận chính để xây dựng các thuật toán rút gọn thuộc tính gồm có mô hình filter, wrapper và embed thuộc tính. Tiếp cận filter là tiếp cận được sử dụng rộng rãi nhất, do các thuật toán được thiết kế theo phương pháp Heuristic nên thời gian thực hiện của thuật toán có thể chấp nhận được trong các tập dữ liệu thực tế. Tiếp cận Wrapper thường được gắn với một mô hình phân lớp cụ thể để đánh giá tập rút gọn ứng viên, do đó tập rút gọn thu được thường có độ chính xác phân lớp tốt nhất. Tuy nhiên với C thuộc tính ta cần đánh giá tới 2^C tập rút gọn ứng viên, do đó cách tiếp cận này có chi phí tính toán rất lớn. Tiếp cận embed thường được nhúng trên các mô hình phân lớp có chọn lọc thuộc tính như mô hình cây quyết định. Gần đây, một số mô hình lai ghép filter và wrapper được nhiều nhà nghiên cứu phát triển do có khả năng tối ưu về thời gian thực hiện của thuật toán và độ chính xác phân lớp cho tập rút gọn thu được. Theo tiếp cận này, quá trình wrapper chỉ phải thực hiện trên các tập rút gọn ứng viên có số lượng nhỏ.

Để xây dựng các thuật toán rút gọn thuộc tính, mô hình chung cho các thuật toán gồm có hai thành phần chính như sau:

- Tiêu chuẩn chọn lọc thuộc tính: bao gồm các phương pháp đánh giá độ quan trọng của thuộc tính như dựa trên độ đo được định nghĩa hay cấu trúc tôpô được định nghĩa.

- Phương pháp tìm kiếm: chủ yếu dựa vào tiếp cận Heuristic như tìm kiếm thuộc tính quan trọng dựa trên tập thuộc tính ban đầu, dựa trên tập thuộc tính lõi hay xuất phát từ tập rỗng.

Các thuộc tính điều kiện trong bảng quyết định có thể được chia làm 03 nhóm có tính chất như sau:

- Tính độc lập (Independent): Bao gồm các thuộc tính điều kiện không tương quan với các thuộc tính điều kiện khác nhưng tương quan với thuộc tính quyết định.

- Tính dư thừa (Redundant): Bao gồm các thuộc tính điều kiện có tương quan với các thuộc tính điều kiện khác nhưng không tương quan với thuộc tính quyết định.

- Tính không phù hợp: Bao gồm các thuộc tính điều kiện không tương quan với các thuộc tính điều kiện khác mà cũng không tương quan với thuộc tính quyết định.

Bên cạnh phương pháp rút gọn thuộc tính truyền thống đã được phát triển hơn ba thập kỉ vừa qua, trong những năm gần đây nhiều nhà nghiên cứu đề xuất cách tiếp cận rút gọn thuộc tính theo tiếp cận tôpô đại số, cách tiếp cận này nhận được nhiều sự quan tâm của cộng động các nhà nghiên cứu lý thuyết về tập thô. Đặc biệt, bài toán rút gọn thuộc tính có liên quan đến khái niệm bất biến của không gian tôpô dưới góc nhìn của đại số trừu tượng.

1.2. Các khái niệm cơ bản

1.2.1. Hệ thông tin và mô hình tập thô truyền thống

Hệ thông tin là công cụ biểu diễn tri thức dưới dạng một bảng dữ liệu gồm A cột ứng với A thuộc tính và U hàng ứng với U đối tượng. Một cách hình thức, hệ thông

tin được định nghĩa như sau.

Định nghĩa 1.1 (Hệ thông tin [14])**.** Hệ thông tin là một bộ tứ $IS = (U, A, V, f)$ trong đó U là tập hữu hạn khác rỗng các đối tượng, A là tập hữu hạn khác rỗng các thuộc tính, $V = \bigcup_{a \in A} V_a$ với V_a là tập giá trị của thuộc tính $a \in A$ và $f : U \times A \to V_a$ là hàm thông tin, $\forall a \in A,\ u \in U$ ta có $f(u, a) \in V_a$.

Một lớp đặc biệt của các hệ thông tin có vai trò quan trọng trong nhiều ứng dụng là bảng quyết định (Decision Table - DT). Bảng quyết định là một hệ thông tin DT với tập thuộc tính A được chia thành hai tập khác rỗng rời nhau C và D , lần lượt được gọi là tập thuộc tính điều kiện và tập thuộc tính quyết định. Tức là $DT = (U, C, D, f)$ với $C \cap D = \emptyset$.

Xét bảng quyết định $DT = (U, C, D, f)$, giả thiết $\forall u \in U,\ \forall d \in D$ đầy đủ giá trị, nếu tồn tại $u \in U$ và $c \in C$ sao cho $c(u)$ thiếu giá trị thì DT được gọi là bảng quyết định không đầy đủ, trái lại DT được gọi là bảng quyết định đầy đủ.

Định nghĩa 1.2 (Quan hệ tương đương [14])**.** Xét bảng quyết định $DT = (U, C, D, f)$. Khi đó, quan hệ của các đối tượng trong U trên tập thuộc tính $B \subseteq C$ ký hiệu bởi $ID(B)$, được định nghĩa bởi:

$$ID(B) = \{(u, v) \in U \,|\, \forall a \in B,\ a(u) = a(v)\} \tag{1.1}$$

Rõ ràng $ID(B)$ là một quan hệ tương đương trên U. Nếu $(u, v) \in ID(P)$ thì hai đối tượng u và v không phân biệt được bởi các thuộc tính trong P. Quan hệ tương đương $ID(B)$ xác định một phân hoạch trên U, ký hiệu là $U/ID(B)$ hay U/B. Ký hiệu lớp tương đương trong phân hoạch U/P chứa đối tượng u là $[u]_B$, khi đó $[u]_B = \{v \in U \,|\, (u, v) \in ID(B)\}$.

Định nghĩa 1.3 (Phân hoạch của thuộc tính [37, 18])**.** Cho bảng quyết định $DT = (U, C, D, f)$ và $P, Q \subseteq C$. Khi đó:

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

2. Phân hoạch U/P được gọi là mịn hơn phân hoạch U/P hay $U/P \preceq U/Q$ khi và chỉ khi $\forall u \in U,\ [u]_P \subseteq [u]_Q$

Định nghĩa 1.4 (Mô hình tập thô truyền thống [14, 37, 18])**.** Trong mô hình lý thuyết tập thô truyền thống, để biểu diễn tập $X \subseteq U$ trên cơ sở tri thức của tập thuộc tính B theo khái niệm tập thô, Pawlack sử dụng hai phép toán xấp xỉ dựa trên các lớp tương đương của U/B. Các phép toán này được gọi là B-xấp xỉ dưới và B-xấp xỉ trên của X trên U/B, ký hiệu lần lượt là $\underline{B}(X)$ và $\overline{B}(X)$. Trong đó:

$$\underline{B}(X) = \{u \in U \,|\, [u]_B \subseteq X\} \tag{1.2}$$

$$\overline{B}(X) = \{u \in U \,|\, [u]_B \cap X \neq \emptyset\} \tag{1.3}$$

Khi đó, $\underline{B}(X)$ là tập các phần tử trong U chắc chắn thuộc X (xác định thuộc), còn $\overline{B}(X)$ là tập các phần tử của U có thể thuộc X dựa trên tập thuộc tính B. Trên cơ sở đó, các tập *không xác định* và tập *xác định không thuộc* được định nghĩa như sau:

$$BN_B(X) = \overline{B}(X) - \underline{B}(X) \tag{1.4}$$

$$U - \overline{B}(X) \tag{1.5}$$

Trong đó: $BN_B(X)$ được gọi là *miền biên của* X theo B, là tập các đối tượng có thể thuộc hoặc không thuộc (không xác định) trong X, còn $U - \overline{B}(X)$ là *miền ngoài* của X là tập các đối tượng chắc chắn không thuộc (*không xác định thuộc*) trong X. Trong trường hợp $BN_B(X) = \emptyset$ thì X được gọi là tập xác định, ngược lại X được gọi là tập thô (Rough Set - RS).

Mệnh đề 1.1 (Các tính chất cơ bản của tập thô [14, 18])**.** *Cho bảng quyết định* $DS = (U, C, D, f)$*, với* $X, Y \subseteq U$ *và* $A \subseteq C$*. Khi đó:*

1. $\underline{A}(\varnothing) = \bar{A}(\varnothing)$, $\underline{A}(U) = \bar{A}(U)$*;*

2. $\underline{A}(X) \subseteq X \subseteq \bar{A}(X)$*;*

3. $\underline{A}(X \cup Y) \supseteq \underline{A}(X) \cup \underline{A}(Y)$;

4. $\underline{A}(X \cap Y) = \underline{A}(X) \cap \underline{A}(Y)$;

5. $\bar{A}(X \cup Y) = \bar{A}(X) \cup \bar{A}(Y)$;

6. $\bar{A}(X \cap Y) \subseteq \bar{A}(X) \cap \bar{A}(Y)$;

7. $\underline{A}(U - X) \subseteq U - \bar{A}(X)$;

8. $\bar{A}(U - X) \subseteq U - \underline{A}(X)$;

9. $\underline{A}\left(\underline{A}(X)\right) = \bar{A}\left(\underline{A}(X)\right) = \underline{A}(X)$;

10. $\bar{A}\left(\bar{A}(X)\right) = \underline{A}\left(\bar{A}(X)\right) = \bar{A}(X)$

1.2.2. Tập thô mờ trực cảm

Định nghĩa 1.5 (Tập mờ [54])**.** Cho U là tập hữu hạn khác rỗng các đối tượng, tập mờ A xác định trên U có dạng: $A = \{\langle x,\ \mu_A\rangle | x \in U\}$. Với $\mu_A : U \to [0,\ 1]$ thỏa mãn $0 \leq \mu_A(x) \leq 1$. Trong đó $\mu_A(x)$ được gọi là độ thành viên của phần tử x trong A.

Mệnh đề 1.2 (Tính chất và phép toán cơ bản của FS[54])**.** *Cho* $A, B \subseteq F(U)$ *với* $F(U)$ *là họ các tập mờ trên* U*. Khi đó:*

1. $A = B$ *nếu* $A(x) = B(x),\ \forall x \in U$

2. $A \subseteq B$ *nếu* $A(x) \leq B(x),\ \forall x \in U$

3. $(A \cup B)(x) = max\{A(x), B(x)\}, \forall x \in U$

4. $(A \cap B)(x) = min\{A(x), B(x)\}, \forall x \in U$

5. $\bar{A}(x)\ = 1 - A(x),\ \forall x \in U$

Để phát triển các phép toán trong môi trường dữ liệu mờ, các toán tử logic rõ cũng được mở rộng cho môi trường dữ liệu mờ. Bảng 1.1 mô tả chi tiết một số toán tử *chuẩn* T-norm và *đối chuẩn* T-conorm tương ứng cho phép toán hợp và giao. Bảng 1.2 mô tả các chi tiết các toán tử kéo theo I-norm và I-conorm. Trong đó : $x = \sqrt{1-a^2}, y = \sqrt{1-b^2}$, $p = \sqrt{2a-a^2}, q = \sqrt{2b-b^2}$.

Dựa trên nền tập mờ, để biểu diễn mối quan hệ của các đối tượng trong cùng một

Bảng 1.1: *Các toán tử T-chuẩn và T-đối chuẩn*

T-norm	T-conorm
$T_m(a,b) = \min\{a,b\}$	$S_m(a,b) = \max\{a,b\}$
$T_p(a,b) = ab$	$S_p(a,b) = a+b-ab$
$T_L(a,b) = \max\{a+b-1,0\}$	$S_L(a,b) = \min\{a+b,1\}$
$T_{\cos}(a,b) = \max\{ab-xy,0\}$	$S_{\cos}(a,b) = \min\{a+b-ab+pq,1\}$

Bảng 1.2: *Các toán tử kéo theo chuẩn và đối chuẩn*

I-norm	I-conorm
$\theta_m(a,b) = \begin{cases} 1, & a \le b \\ b, & a > b \end{cases}$	$\sigma_m(a,b) = \begin{cases} 0, & a \ge b \\ b, & a < b \end{cases}$
$\theta_p(a,b) = \begin{cases} 1, & a = 0 \\ \min\left\{1, \frac{b}{a}\right\}, & \text{otherwise} \end{cases}$	$\sigma_p(a,b) = \begin{cases} 1, & a = 0; \\ \max\left\{0, \frac{b-a}{1-a}\right\}, & \text{otherwise.} \end{cases}$
$\theta_L(a,b) = \min\{b-a+1,1\}$	$\sigma_L(a,b) = \min\{0,b-a\}$
$\theta_{\cos}(a,b) = \begin{cases} 1, & a \le b \\ ab+xy, & a > b \end{cases}$	$\sigma_{\cos}(a,b) = \begin{cases} 0, & a > b \\ a+b-ab-pq, & a \le b \end{cases}$

tập được trơn hơn, phản ánh rõ nét độ tương tự giữa các đối tượng, đặc biệt là quan hệ giữa các đối tượng thuộc $\mathbb{R}$, người ta sử dụng quan hệ tương đương mờ thay cho quan hệ tương đương truyền thống, được định nghĩa như sau:

Định nghĩa 1.6 (Quan hệ tương đương mờ [55])**.** Xét R là quan hệ tương đương trên U không rỗng. Khi đó R được gọi là quan hệ tương đương mờ nếu các tiêu chuẩn sau đây được thỏa mãn:

1. Có tính phản xạ: nếu $R(x,x) = 1$ với mọi$x \in U$;
2. Có tính đối xứng nếu $R(x,y) = R(y,x)$ với mọi $x,y \in U$;
3. Có tính bắc cầu nếu $T(R(x,y),R(y,z)) \le R(x,z)$ với mọi $x,y,z \in U$.

Định nghĩa 1.7 (Ma trận quan hệ)**.** Cho R là một quan hệ tương mờ trên U, khi đó quan hệ giữa các đối tượng trong U theo R có thể được biểu diễn bởi ma trận quan hệ $M = [i,j]_{|U|\times|U|}$.

Đây là ma trận vuông có kích thước $|U| \times |U|$, trong đó $|U|$ là số các đối tượng trong U và i, j là chỉ số của phần tử tại hàng i và cột j trên ma trận M. Khi đó mỗi giá

trị tại hàng i cột j cho biết độ tương tự giữa đối tượng i và j trong U.

Để có thể mô tả thông tin đầy đủ hơn trong các trường hợp dữ liệu phức tạp như thông tin về phiếu tín nhiệm của một ứng viên cần được đánh giá. Trong đó có các thành phần về tỉ lệ tín nhiệm và tỉ lệ bất tín nhiệm. Khi đó khái niệm về tập mờ trực cảm ra đời và được định nghĩa như sau:

Định nghĩa 1.8 (Tập mờ trực cảm [56])**.** Cho U là tập không rỗng các đối tượng, tập mờ trực cảm X trên U được xác định bởi:

$$X = \{\langle x, \mu_X(x), \nu_X(x)\rangle \,|x \in U\} \tag{1.6}$$

Trong đó, $\mu_X(x) \in [0,1]$ là mức độ thành viên của $x \in U$ với X và $\nu_A(x) \in [0,1]$ là mức độ không thành viên của $x \in U$ với X sao cho $0 \leq \mu_X(x) + \nu_X(x) \leq 1 \forall x \in U$.

Khi đó, với mỗi tập mờ Y truyền thống, tập mờ trực cảm X có thể được xác định bởi:

$$X = \{\langle x, \mu_Y(x), 1 - \mu_Y(x)\rangle \,|x \in U\} \tag{1.7}$$

Nếu $0 \leq \mu_X(x) + \nu_X(x) < 1$ thì $\pi_X(x) = 1 - \mu_X(x) - \nu_X(x)$ được gọi là độ do dự của $x \in U$ với X.

Định nghĩa 1.9 (Phép toán cơ bản của IFS [36])**.** Xét P và Q là các tập mờ trực cảm xác định trên U. Khi đó hợp và giao của của P và Q được xác định như sau:

$$P \cup Q = \{\langle x, \vee(\mu_P(x), \mu_Q(x)), \wedge(\nu_P(x), \nu_Q(x))\rangle\} \tag{1.8}$$

$$P \cap Q = \{\langle x, \wedge(\mu_P(x), \mu_Q(x)), \vee(\nu_P(x), \nu_Q(x))\rangle\} \tag{1.9}$$

Mệnh đề 1.3 (Quan hệ của hai IFS [36])**.** *Xét P và Q là các tập mờ trực cảm xác định trên U:*

1. *$P \subseteq Q$ khi và chỉ khi $\mu_P(x) \leq \mu_Q(x)$ và $\nu_Q(x) \geq \nu_P(x)$ với mọi $x \in U$*
2. *$P = Q$ khi và chỉ khi $P \subseteq Q$ và $Q \subseteq P$.*

Dựa trên nền tập mờ trực cảm, để biểu diễn mối quan hệ của các đối tượng trong cùng một tập được chặt hơn. Đặc biệt là quan hệ giữa các đối tượng thuộc $\mathbb{R}$ trên các miền giá trị có chứa nhiễu. Khi đó, quan hệ tương đương của mô hình tập thô truyền thống được mở rộng trên nền tập mờ trực cảm bằng quan hệ tương đương mờ trực cảm theo định nghĩa sau đây:

Định nghĩa 1.10 (Quan hệ tương đương mờ trực cảm [36])**.** Xét quan hệ R xác định trên U không rỗng. Khi đó R được gọi là quan hệ tương đương mờ trực cảm nếu:

1. Có tính phản xạ: $\mu_{R(x,x)} = 1$ và $\nu_{R(x,x)} = 0$ với mọi $x \in U$;

2. Có tính đối xứng: $\mu_{R(x,y)} = \mu_{R(y,x)}$ và $\nu_{R(x,y)} = \nu_{R(y,x)}$ với mọi $x, y \in U$;

3. Có tính bắc cầu: $\mu_{R(x,z)} \geq \bigvee\limits_{y \in U} \left[\mu_{R(x,y)} \wedge \mu_{R(y,z)}\right]$ và $\nu_{R(x,z)} \leq \bigwedge\limits_{y \in U} \left[\nu_{R(x,y)} \vee \nu_{R(y,z)}\right]$ với mọi $x, y \in U$.

Định nghĩa 1.11 (Lực lượng của một tập mờ trực cảm [36])**.** Cho tập mờ trực cảm X xác định trên U, với U là tập không rỗng các đối tượng. Khi đó lực lượng của X được xác định như sau:

$$|X| = \sum_{i=1}^{|U|} \frac{1 + \mu_i - \nu_i}{2} \tag{1.10}$$

Dựa trên quan hệ tương đương mờ trực cảm và các toán tử logic mờ, mô hình tập thô mờ trực cảm được mở rộng theo định nghĩa sau:

Định nghĩa 1.12 (Mô hình tập thô mờ trực cảm [36])**.** Cho bảng quyết định $DT = (U, C, D, f)$, R là quan hệ tương đương mờ xác định trên U và $A \subseteq U$, ta có:

$$\underline{A}(x) = \bigwedge_{y \in U} I(R(x,y), A(y)) \tag{1.11}$$

$$\bar{A}(x) = \bigvee_{y \in U} T(R(x,y), A(y)) \tag{1.12}$$

Trong đó T và I tương ứng với toán tử $T-norm$ và toán tử kéo theo $I-norm$ được chọn trong Bảng 1.1 và Bảng 1.2.

1.2.3. Không gian tôpô

Không gian tôpô [38] được kí hiệu bởi cặp (U,τ), trong đó U là tập không rỗng các đối tượng và τ là họ các tập con của U thỏa mãn các điều kiện sau:

1. $\Phi \in \tau$ and $U \in \tau$.

2. $\forall G_1, G_2 \in \tau$, luôn tồn tại $G_3 \in \tau$ sao cho $G_3 = G_1 \cap G_2$.

3. $\forall G_1, G_2 \in \tau$, luôn tồn tại $G_3 \in \tau$ sao cho $G_3 = G_1 \cup G_2$.

Cặp (U,τ) được gọi là không gian tôpô xác định trên U với các phần tử là các tập mở và là tập con của U, phần bù của các tập mở được gọi là các tập đóng.

Định nghĩa 1.13 (Cơ sở (base) [57]). Cho U là tập không rỗng các đối tượng. Khi đó cơ sở (base) của tôpô τ trên U là họ các tập con của U kí hiệu là B sao cho:

1. Với mỗi $x \in U$, tồn tại $G \subseteq B$ sao cho $x \in G$.

2. Vơi mọi $G_1, G_2 \in B$, nếu $x \in G_1 \cap G_2$, thì tồn tại $G_3 \in B$ sao cho $x \in G_3$.

Định nghĩa 1.14 (Cơ sở con (subbase) [57]). Cho không gian tôpô (U,τ). Khi đó $S \subseteq \tau$ được gọi là cơ sở con (subbase) của tôpô τ nếu giao hữu hạn các tập con của S tạo thành cơ sở B của tôpô τ.

Định nghĩa 1.15 (Tôpô Hausdorff [38]). Cho không gian xấp xỉ (U,τ), tôpô τ được gọi là tôpô Hausdorff nếu mọi $x \neq y \in \tau$ luôn tồn tại hai lân cận mở $G_x, G_y \in \tau$ tương ứng với x, y sao cho $G_x \cap G_y = \emptyset$.

Để xây dựng tôpô, cách truyền thống là dựa trên cơ sở (base). Trong đó, mỗi phần tử của tôpô tương ứng là hợp các phân tử trong một tập con của cơ sở (base). Trong đó cơ sở base được tạo ra từ cơ sở con (subbase).

Để xác định đối tượng nào trong U có thể thuộc vào tập mục tiêu A với $A \subseteq U$ dựa trên khái niệm tập đóng của tôpô ta sử dụng công thức sau:

$$\bar{A} = \cap\{F \subseteq U : A \subseteq F\} \tag{1.13}$$

Để xác định các đối tượng chắc chắn thuộc vào tập mục tiêu A với $A \subseteq U$ dựa trên

khái niệm tập mở của tôpô ta sử dụng công thức sau:

$$A^{\circ} = \cup\{G \subseteq U : G \subseteq A\} \tag{1.14}$$

Mệnh đề 1.4 (Cấu trúc tôpô theo tiếp cận tập thô [39])**.** *Cho bảng quyết định* $DT = (U, C, D, f)$ *và quan hệ tương đương R xác định trên* U*. Khi đó* $\tau = \{X \subseteq U | \underline{R}(X) = \overline{R}(X)\}$ *là một tôpô trên* U*.*

Định nghĩa 1.16 (Tôpô mờ trực cảm IFT)**.** [20] Cho τ là họ các tập mờ trực cảm xác định trên tập không rỗng U. Khi đó τ được gọi là tôpô mờ trực cảm nếu:

1. 0_{IF}, $1_{IF} \in \tau$
2. $G_1 \cap G_2 \in \tau : G_1, G_2 \in \tau$
3. $\cup G_i \in \tau : \{G_i : G_i \in \tau, i \in I\}$

Khi đó, cặp (U, τ) được gọi là không gian tôpô mờ trực cảm. Trong đó, 0_{IF} và 1_{IF} lần lượt là các tập mờ trực cảm nhỏ nhất và lớn nhất trên U.

1.2.4. Tập rút gọn

Trong bảng quyết định, các thuộc tính điều kiện được phân thành ba nhóm: thuộc tính lõi (core attribute), thuộc tính rút gọn (reductive attribute) và thuộc tính dư thừa (redundant attribute). Thuộc tính lõi là những thuộc tính luôn xuất hiện trong các tập rút gọn. Thuộc tính dư thừa là những thuộc tính mà việc loại bỏ chúng không ảnh hưởng đến việc phân lớp tập dữ liệu. Thuộc tính không liên quan là thuộc tính không mang lại những thông tin đóng góp cho việc ra quyết định. Thuộc tính rút gọn là thuộc tính xuất hiện trong một tập rút gọn nào đó của bảng quyết định.

1.3. Một số công thức tính toán độ thành viên

Bảng quyết định 1.3 được biểu diễn bởi bộ $DT = (U, C, D, f)$. Trong đó $C \cap D = \emptyset$, U là một tập không rỗng các đối tượng, C là tập không rỗng các thuộc tính điều kiện và D là thuộc tính quyết định. Hàm thông tin f_c xác định một giá trị trong V_c tương

ứng với mỗi $u \in U$ và $c \in C$, trong đó V_c là miền giá trị của thuộc tính c và V_c thuộc $\mathbb{R}$. Hàm thông tin f_D xác định một giá trị trong V_D với mỗi $u \in U$, trong đó V_D là miền giá trị của D và V_D thuộc $\mathbb{N}$.

Bảng 1.3: *Mô tả cấu trúc bảng quyết định số*

U	a	b	c	d	e	f	D
u_1	1.0	0.4	0.8	0.2	1.0	0.0	0
u_2	1.0	0.4	0.2	0.4	0.2	0.8	1
u_3	0.8	0.6	1.0	0.0	0.6	0.4	0
u_4	0.2	0.6	0.8	0.2	0.0	1.0	1
u_5	0.2	0.8	0.8	0.2	0.0	1.0	1
u_6	0.2	0.8	0.2	0.8	0.0	1.0	0

1.3.1. Chuẩn hóa dữ liệu

Đối với các bảng quyết định có thuộc tính điều kiện miền giá trị số, các thuộc tính thường được chuẩn hóa để tăng hiệu quả huấn luyện cho các mô hình. Theo các kết quả khảo sát của [55] Sau đây là một số phương pháp chuẩn hóa dữ liệu được sử dụng phổ biến:

1. Min-max normalization:

$$F\left(f_{c_k}(x_i)\right) = \frac{f_{c_k}(x_i) - min_{c_k}}{max_{c_k} - min_{c_k}} \left(max'_{c_k} - min'_{c_k}\right) + min'_{c_k} \tag{1.15}$$

Trong đó max_{c_k} và min_{c_k} là các giá trị nhỏ nhất và lớn nhất của thuộc tính c_k. Sau khi chuẩn hóa, các giá trị của thuộc tính được đưa về đoạn mới $[min'_{c_k}, max'_{c_k}]$.

2. z-score normalization:

$$F\left(f_{c_k}(x_i)\right) = \frac{f_{c_k}(x_i) - \overline{c_k}}{\sigma_{c_k}} \tag{1.16}$$

Trong đó, $\overline{c_k}$ và σ_{c_k} kí hiệu là giá trị trung bình và độ lệch chuẩn của thuộc tính c_k.

3. Chuẩn hóa về thang đo hệ 10:

$$F\left(f_{c_k}(x_i)\right) = \frac{f_{c_k}(x_i)}{10^l} \tag{1.17}$$

Trong đó I là số nguyên nhỏ nhất sao cho $\max\left(\left|F\left(f_{c_k}(x_i)\right)\right|\right) < 1$.

1.3.2. Độ đo độ tương tự

Cho bảng quyết định $DT = (U, C, D, f)$ với $B \subseteq C$ và quan hệ tương đương mờ R. Khi đó R sẽ chia U thành các lớp tương đương mờ theo B gọi là phân hoạch mờ của B trên U kí hiệu là U/R_B. Trong đó $U/R_B = \left\{[x_1]_{R_B}, [x_2]_{R_B}, \ldots, [x_n]_{R_B}\right\}$, với: $[x_i]_{R_B} = \left(r_{i1}^B, r_{i2}^B, \ldots, r_{in}^B\right)$. Rõ ràng, $[x_i]_{R_B}$ là một tập mờ trên R_B. ta có $[x_i]_{R_B}(x_j) = R_B(x_i, x_j) = r_{ij}^B$. Nếu $R_B(x_i, x_j) = 1$, nghĩa là x_j chắc chắn thuộc $[x_i]_{R_B}$; Nếu $R_B(x_i, x_j) = 0$, thì x_j chắc chắn không thuộc $[x_i]_{R_B}$.

Lực lượng của tập mờ $[x_i]_{R_B}$ được xác định bởi $\left|[x_i]_{R_B}\right| = \sum_{j=1}^{n} R_B(x_i, x_j)$. Ta có $1 \leq \left|[x_i]_{R_B}\right| \leq n$ với $n = |U|$. Để xác định độ tương tự r_{ij}^B. Sau đây là một số công thức tính độ tương tự được sử dụng phổ biến.

1. Khoảng cách [58] được xác định bởi

$$r_{ij}^B = 1 - \frac{1}{\sqrt[p]{C}} \Delta_p^B(x_i, x_j) \tag{1.18}$$

Trong đó $h = |B|$, $\Delta_p^B(x_i, x_j) = \sqrt[p]{\sum_{k=1}^{h}\left(f_{c_k}(x_i) - f_{c_k}(x_j)\right)^p}$. Khi $p = 1, p = 2$, và $p = \infty, \Delta_p^B(x_i, x_j)$ tương ứng với khoảng cách của Manhattan, Euclidean, và Chebyshev. C là hệ số khoảng cách sao cho r_{ij}^B thuộc đoạn $[0, 1]$.

2. Độ tương quan [59] được xác định bởi:

$$r_{ij}^B = \frac{\sum_{k=1}^{h}\left|f_{c_k}(x_i) - \overline{x_i}\right|\left|f_{c_k}(x_j) - \overline{x_j}\right|}{\sqrt{\sum_{k=1}^{h}\left(f_{c_k}(x_i) - \overline{x_i}\right)^2}\sqrt{\sum_{k=1}^{h}\left(f_{c_k}(x_j) - \overline{x_j}\right)^2}} \tag{1.19}$$

Trong đó: $\overline{x_i} = \frac{1}{h}\sum_{c=1}^{h} f_{c_k}(x_i), \overline{x_j} = \frac{1}{h}\sum_{c=1}^{h} f_{c_k}(x_j)$.

2. Độ phân li [60] được xác định bởi:

$$r_{ij}^B = \wedge_{c=1}^{h} r_{ij}^{c_k} = \min_{c=1}^{h} r_{ij}^{c_k} \tag{1.20}$$

4. Nhân đại số [61] được xác định bởi:

$$r_{ij}^{B} = \prod_{c=1}^{h} r_{ij}^{c_k} \tag{1.21}$$

5. Phương pháp T-norm [62] được xác định bởi

$$r_{ij}^{B} = \underset{c=1}{T} r_{ij}^{c_k} \tag{1.22}$$

6. Hàm Kernel [63, 64]:

(i) Gaussian kernel:

$$r_{ij}^{B} = \exp\left(-\frac{\|x_i - y_i\|_B^2}{2\delta^2}\right) \tag{1.23}$$

(ii) Exponential kernel:

$$r_{ij}^{B} = \exp\left(-\frac{\|x_i - y_i\|_B}{\delta}\right) \tag{1.24}$$

(iii) Rational quadratic kernel:

$$r_{ij}^{B} = 1 - \frac{\|x_i - y_i\|_B^2}{\|x_i - y_i\|_B \|x_i - y_i\|_B^2 + \delta} \tag{1.25}$$

(iv) Spherical kernel:

$$r_{ij}^{B} = 1 - \frac{3}{2}\frac{\|x_i - y_i\|_B}{\delta} + \frac{1}{2}\left(\frac{\|x_i - y_i\|_B}{\delta}\right)^3 \tag{1.26}$$

với $\|x_i - y_i\|_B < \delta$

(v) Circular kernel:

$$r_{ij}^{B} = \frac{2}{\pi}\arccos\left(\frac{\|x_i - y_i\|_B}{\delta}\right) - \frac{2}{\pi}\frac{\|x_i - y_i\|_B}{\delta}\sqrt{1 - \left(\frac{\|x_i - y_i\|_B}{\delta}\right)^2} \tag{1.27}$$

với $\|x_i - y_i\|_B < \delta$

Trong đó δ là tham số của hàm kernel.

7. Một số phương pháp khác [65]:

$$r_{ij}^{B} = \frac{1}{|B|}\left|\left\{c_k \in B \mid f_{c_k}(x_i) = f_{c_k}(x_j)\right\}\right| \tag{1.28}$$

Đối với các thuộc tính có giá trị rời rạc, độ thành viên $r_{ij}^{c_k}$ được xác định như sau:

$$r_{ij}^{c_k} = \begin{cases} 1, & \text{nếu } f_{c_k}(x_i) = f_{c_k}(x_j) \\ 0, & \text{còn lại.} \end{cases} \tag{1.29}$$

Đối với các thuộc tính có giá trị số, $r_{ij}^{c_k}$ có thể được xác đinh bởi hàm F như sau:

$$r_{ij}^{c_k} = F(x_i, x_j) \tag{1.30}$$

Trong đó, F thỏa mãn $F(x_i,x_i) = 1, F(x_i,x_j) = F(x_j,x_i)$, và $F(x_i,x_j) \in [0,1]$.

Sau đây là một số ví dụ của hàm F

(1) $r_{ij}^{c_k} = 1 - \left|f_{c_k}(x_i) - f_{c_k}(x_j)\right|$.

(2) $r_{ij}^{c_k} = \max\left(\min\left(\frac{f_{c_k}(x_j) - f_{c_k}(x_i) + \sigma_{c_k}}{\sigma_{c_k}}, \frac{f_{c_k}(x_i) - f_{c_k}(x_j) + \sigma_{c_k}}{\sigma_{c_k}}\right), 0\right)$

Trong đó, σ_{c_k} được gọi là độ lệch chuẩn.

1.4. Phương pháp đánh giá tập rút gọn

1.4.1. Các tiêu chí đánh giá

Các thuật toán rút gọn thuộc tính theo tiếp cận độ đo hiện nay thường được đánh giá dựa trên ba tiêu chí gồm có: *kích thước* của tập rút gọn thu được, *độ chính xác phân lớp* của tập rút gọn trên mô hình được huấn luyện và *thời gian thực hiện* của thuật toán.

Tập rút gọn thu được từ thuật toán có kích thước càng nhỏ thì càng hiệu quả về thời gian xây dựng mô hình. Độ chính xác càng cao thì càng khẳng định được phương pháp chọn lọc thuộc tính và cấu trúc tập rút gọn thu được hiệu quả. Thời gian thực hiện càng nhanh cho biết khả năng rút gọn dữ liệu của thuật toán trên các tập dữ liệu

lớn.

Mục tiêu chung của các thuật toán rút gọn thuộc tính là cố gắng đạt được cả ba tiêu chí trên, tuy nhiên trong thực tế với các bộ dữ liệu nhiễu và phức tạp. Tiêu chí kích thước và độ chính xác phân lớp của tập rút gọn được nhiều nhà nghiên cứu quan tâm. Sau đây là một số độ đo đánh giá khả năng phân lớp chính xác của mô hình trên các tập rút gọn.

1.4.2. Mô hình và dữ liệu đánh giá

Theo khảo sát của các tác giả trong công trình [55] cho thấy các thuật toán phân lớp được sử dụng phổ biến trong đánh giá độ chính xác phân lớp của các tập dữ liệu trước và sau khi rút gọn gồm có: mô hình cây quyết định C.45, cây phân lớp và hồi quy CART, máy vector hỗ trợ SVM và mô hình phân lớp lân cận k-NN. Đối với các bảng quyết định có thuộc tính miền giá trị số, mô hình phân lớp k-NN và SVM được sử dụng nhiều hơn các mô hình phân lớp còn lại.

Hầu hết các thuật toán rút gọn thuộc tính được nghiên cứu và đánh giá dựa trên các tập dữ liệu được tải về từ UCI. Đây là kho dữ liệu đa dạng các chủ đề, đáng tin cậy. Được nhiều chuyên gia và các nhà nghiên cứu sử dụng.

1.4.3. Chỉ số đánh giá

Để đánh giá hiệu quả về độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán. Nhiều chỉ số đánh giá đã được đề xuất, trong đó các chỉ số này đều sử dụng ma trận lầm lẫn kết hợp với phương pháp đánh giá chéo [55].

Tiếp cận đánh giá chéo là phương pháp đánh giá quan trọng trong các mô hình của học máy, trong đó tập dữ liệu ban đầu được chia thành hai phần chính là tập dữ liệu huấn luyện và tập dữ liệu kiểm thử. Có hai loại đánh giá chéo phổ biến gồm có: phương pháp k-fold, phương pháp left-one-out. Trong hai loại phương pháp đánh giá chéo này, phương pháp k-fold được sử dụng rộng rãi nhất trong các nghiên cứu về mô hình học máy. Phương pháp này thực hiện việc chia ngẫu nhiên tập dữ liệu ban đầu

thành k phần trong đó $k-1$ phần được huấn luyện và 1 phần được dùng để đánh giá. Thực hiện việc thay đổi tỉ lệ này với k lần khác nhau ta được phương pháp đánh giá chéo k-fold.

Bảng 1.4: *Ma trận lầm lẫn nhị phân*

Actual class	Predicted class	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Ma trận lầm lẫn là một công cụ hiệu quả trong việc phân tích hiệu năng của các bộ phân lớp dữ liệu. Với mô hình phân lớp dữ liệu nhị phân, ma trận lầm lẫn có thể được biểu diễn trong Bảng 1.4. Sau đây là một số chỉ số để đánh giá độ chính xác phân lớp dựa trên ma trận lầm lẫn.

1. Accuracy:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}. \tag{1.31}$$

2. Error:

$$Error = \frac{FP+FN}{TP+TN+FP+FN} \tag{1.32}$$

3. Precision:

$$Precision = \frac{TP}{TP+FP} \tag{1.33}$$

4. Recall:

$$Recall = \frac{TP}{TP+FN} \tag{1.34}$$

5. F measure (F) :

$$F = \frac{2\times \text{ precision } \times \text{ recall}}{\text{precision } + \text{ recall}} \tag{1.35}$$

6. F_β measure $\left(F_\beta\right)$:

$$F_\beta = \frac{\left(1+\beta^2\right) \times \text{ precision } \times \text{ recall}}{\beta^2 \times \text{ precision } + \text{ recall}} \tag{1.36}$$

1.5. Một số phương pháp rút gọn thuộc tính

1.5.1. Phương pháp rút gọn thuộc tính theo tiếp cận ma trận phân biệt

Vào năm 1992, Skowron và Rauszer lần đầu tiên giới thiệu phương pháp rút gọn thuộc tính theo tiếp cận ma trận phân biệt trên nền tập thô [66]. Khi đó ma trận phân biệt có kích thước $n \times n$ với $n = |U|$, kí hiệu là $M(DS) = (c_{ij})_{n\times n}$ được xác định bởi:

$$c_{ij} = \begin{cases} \left\{c \in C \mid c\left(x_i\right) \neq c\left(x_j\right)\right\}, \omega\left(x_i, x_j\right) \\ \emptyset, \text{ còn lại.} \end{cases} \tag{1.37}$$

Trong đó: $\omega\left(x_i, x_j\right)$ thỏa mãn một trong các điều kiện sau đây:

1. $x_i \in \text{POS}_C(D) \wedge x_j \notin \text{POS}_C(D)$;

2. $x_i \notin POS_C(D) \wedge x_j \in POS_C(D)$;

3. $x_i, x_j \in POP_C(D) \wedge (x_i, x_j) \notin \text{IND}(D)$.

Hàm phân biệt của ma trận phân biệt $f(C,D)$ là một hàm Boolean được xác định như sau:

$$f(C,D) = \wedge\left\{\vee c_{ij} \mid c_{ij} \neq \emptyset\right\} \tag{1.38}$$

Khi đó tập thuộc tính lõi được xác định bởi:

$$\text{core}_C(D) = \left\{c \mid c_{ij} = \{c\}\right\} \tag{1.39}$$

Năm 2008, Tsang và các cộng sự [67] giới thiệu ma trận phân biệt mờ cho mô hình

rút gọn thuộc tính dựa trên công thức xây dựng ma trận quan hệ sau:

$$c_{ij} = \begin{cases} \left\{c \in C \mid 1 - R_c\left(x_i, x_j\right) \geq \lambda_i\right\}, \lambda_j < \lambda_i; \\ \emptyset, \text{ còn lại,} \end{cases} \tag{1.40}$$

Trong đó: $\lambda_i = R_C\left[x_i\right]_D\left(x_i\right)$, và $\lambda_j = R_C\left[x_i\right]_D\left(x_j\right)$.

Năm 2009 Jensen và các cộng sự [68] định nghĩa lại ma trận của Tsang dựa theo công thức:

$$c_{ij} = \left\{c_\mu \mid \mu = N\left(R_c\left(x_i, x_j\right)\right)\right\} \tag{1.41}$$

Tuy nhiên, các ma trận phân biệt này còn sinh ra quá nhiều tập rút gọn ứng viên do đó các tác giả [69] đề xuất ma trận phân biệt các thuộc tính điều kiện ràng buộc bởi thuộc tính quyết định như sau:

$$c_{ij} = \begin{cases} \left\{c \in C \mid T\left(R_c\left(x_i, x_j\right), \lambda_i\right) = 0\right\}, \text{ if } x_j \notin \left[x_i\right]_D; \\ \emptyset, \text{ còn lại} \end{cases} \tag{1.42}$$

trong đó $\lambda_i = R_{\theta C}\left[x_i\right]_D\left(x_i\right)$.

Chen và các cộng sự [70] kết hợp ma trận phân biệt với tập thô mờ để rút gọn thuộc tính trong bảng quyết định hybrid. Ma trận phân biệt được xác định bởi:

$$c_{ij} = \begin{cases} \left\{c \in C \mid \varphi_C\left(x_i\right) - R_D(c)\left(x_i, x_j\right) \leq \varepsilon\right\}, \text{ if } x_j \notin \left[x_i\right]_D \text{ and } \varphi_C\left(x_i\right) \neq 0; \\ \emptyset, \text{ còn lại,} \end{cases} \tag{1.43}$$

Trong đó $\varphi_C\left(x_i\right) = \min_{D(x) \neq D(y)} R_D(c)\left(x_i, x_j\right)$. Nếu $(x, y) \in R_D(c)$, thì $R_D(c)\left(x_i, x_j\right) = 1$; ngược lại $R_D(c)\left(x_i, x_j\right) = 0$. Cho đến nay, có khá nhiều phương pháp rút gọn thuộc tính theo tiếp cận ma trận phân biệt được đề xuất trong các công trình [71, 72, 73, 74].

1.5.2. Phương pháp rút gọn thuộc tính theo tiếp cận độ đo

Hầu hết các phương pháp rút gọn thuộc tính theo tiếp cận độ đo hiện nay đều sử dụng mô hình filter thuộc tính để xác định tập rút gọn. Ba thành phần quan trọng nhất

để xây dựng mô hình filter thuộc tính gồm có:

1) Phương pháp tìm kiếm: Hầu hết các thuật toán rút gọn thuộc tính hiện nay đều sử dụng tiếp cận tìm kiếm tham lam. Trong đó kỹ thuật tìm kiếm tham lam theo chiều tiến thường xuất phát từ tập rút gọn ban đầu, sau đó bổ sung lần lượt vào tập rút gọn từng thuộc tính quan trọng nhất đối với tập rút gọn. Đối với tiếp cận tìm kiếm tham lam lùi, xuất phát từ tập thuộc tính ban đầu, lần lượt loại bỏ đi các thuộc tính dư thừa, không liên quan đến việc ra quyết định.

2) Phương pháp đánh giá: Hầu hết các thuật toán rút gọn thuộc tính theo tiếp cận độ đo đều sử dụng phương pháp bảo toàn độ đo độ để xác định tập rút gọn, dựa trên các chiến lược chọn lọc thuộc tính quan trọng nhất và loại bỏ đi thuộc tính dư thừa cho đến khi thông tin ra quyết định của tập rút gọn là tương đương với tập thuộc tính gốc thì thuật toán kết thúc.

Sau đây là một số độ đo được sử dụng để đánh giá độ quan trọng của thuộc tính và định nghĩa tập rút gọn trong các mô hình rút gọn thuộc tính theo tiếp cận độ đo hiện nay.

1.5.2.1. độ đo độ phụ thuộc

Độ đo độ phụ thuộc được giới thiệu bởi [39] nhận được nhiều quan tâm của các nhà nghiên cứu, cơ sở của độ đo này dựa trên khái niệm miền dương (POS) của tập thô.

Cho bảng quyết định $DT = (U, C, D, f)$ với $B \subseteq C$, $X \subseteq U$ và R là quan hệ tương đương trên U. Khi đó miền dương của D theo B được xác định như sau:

$$\mathrm{POS}_B(D) = \bigcup_{X_i \in U/D} \underline{R_B} X_i \tag{1.44}$$

Khi đó, độ phụ thuộc của D vào B được xác định bởi:

$$\gamma_B(D) = \frac{|P_B(D)|}{|U|} = \frac{\sum_{x \in U} P_B S_B(D)(x)}{|U|} \tag{1.45}$$

Trên cơ sở đó, độ quan trọng của thuộc tính theo tiếp cận POS được xác định dựa trên hai công thức chính sau đây:

$$\mathrm{Sig}_1(a,B,D) = \gamma_B(D) - \gamma_{B-a}(D) \tag{1.46}$$

$$\mathrm{Sig}_2(a,B,D) = \gamma_{B\cup a}(D) - \gamma_B(D) \tag{1.47}$$

Trong đó công thức 1.46 phù hợp với kỹ thuật tìm kiếm tham lam lùi còn công thức 1.47 phù hợp với kỹ thuật tìm kiếm tham lam tiến.

Trên cơ sở đó, các phương pháp rút gọn thuộc tính theo tiếp cận độ phụ thuộc được phát triển dựa trên mở rộng các độ đo này. Chi tiết các phương pháp được trình bày trong Bảng 1.5.

Bảng 1.5: *Tổng hợp phương pháp rút gọn thuộc tính theo độ phụ thuộc*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Tiếp cận	Tập nền	Tiêu chuẩn đánh giá
1	[75, 79, 80, 81, 82, 83, 84, 85, 86, 76, 77, 78]	Hybrid	NRS	Tập cổ điển	accuracy, size, compuation time
2	[27, 32, 87, 88, 89, 90, 91, 92]	Number	NRS	FS	accuracy, size, compuation time
3	[93]	Number	NRS	IFS	accuracy, size, compuation time
4	[94]	Hybrid	PRS	Tập cổ điển	accuracy, size, compuation time
5	[25, 99, 24, 26, 87, 17, 89, 100, 101, 95, 23, 58, 67, 96, 27, 28, 29, 97, 98]	Number	FRS	FS	accuracy, size, compuation time
6	[102, 34, 105, 106, 107, 108, 93, 36, 35, 103, 104]	Number	IFRS	FS	accuracy, size, compuation time

1.5.2.2. độ đo độ chắc chắn

Độ đo độ chắc chắn là một độ đo quan trọng được dùng để đánh giá sự chắc chắn của thông tin trong bảng quyết định. dựa trên khái niệm Entropy thông tin của Shanon, một số độ đo độ chắc chắn được mở rộng cho bài toán rút gọn thuộc tính gồm có:

(1) Entropy điều kiện [57]: Dựa trên khái niệm Entropy thông tin của Shanon, ba loại độ đo được mở rộng để đánh giá độ chắc chắn thông tin gồm có:

- entropy thông tin:

$$FE(B) = -\frac{1}{|U|}\sum_{i=1}^{|U|}\log_2\frac{\left|[x_i]_{R_B}\right|}{|U|} \tag{1.48}$$

- entropy kết hợp:

$$FE(B,E) = -\frac{1}{|U|}\sum_{i=1}^{|U|}\log_2\frac{\left|[x_i]_{R_B}\cap[x_i]_{R_E}\right|}{|U|} \tag{1.49}$$

- entropy có điều kiện:

$$FE(E \mid B) = -\frac{1}{|U|}\sum_{i=1}^{n}\log_2\frac{\left|[x_i]_{R_E}\cap[x_i]_{R_B}\right|}{\left|[x_i]_{R_B}\right|} \tag{1.50}$$

Khi đó $\forall a \in C-B, B \subseteq C$, hai phương pháp tính độ quan trọng của thuộc tính a với tập thuộc tính B được xác định như sau:

$$\mathrm{Sig}(a,B) = FE(B) - FE(B-\{a\}) \tag{1.51}$$

$$\mathrm{Sig}(a,B,D) = FE(D \mid B-\{a\}) - FE(D \mid B) \tag{1.52}$$

(2) Mutual information [93]: Vào năm 2008, An và các cộng sự sử dụng khái niệm cực đại hóa thông tin liên quan và cực tiểu hóa thông tin dư thừa (mRMR) kết hợp với

khái niệm entropy thông tin để đề xuất thuật toán mRMR:

$$\mathrm{FMI}(E;B) = FE(E) - FE(E \mid B) = -\frac{1}{|U|}\sum_{i=1}^{|U|}\log_2 \frac{\left|[x_i]_{R_B}\right| \cdot \left|[x_i]_{R_E}\right|}{|U| \cdot \left|[x_i]_{R_B} \cap [x_i]_{R_E}\right|}. \quad (1.53)$$

Khi đó $\forall a \in C - B, B \subseteq C$, độ quan trọng của thuộc tính a với tập thuộc tính B được xác định như sau:

$$\mathrm{Sig}(a,B,D) = \mathrm{FMI}(B \cup \{a\};D) - \mathrm{FMI}(B;D) \quad (1.54)$$

Trên cơ sở đó, các phương pháp rút gọn thuộc tính theo tiếp cận độ chắc chắn được phát triển dựa trên mở rộng các độ đo này. Chi tiết các phương pháp được trình bày trong Bảng 1.6.

Bảng 1.6: *Tổng hợp phương pháp rút gọn thuộc tính theo độ không chắc chắn*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Tiếp cận	Tập nền	Tiêu chuẩn đánh giá
1	[109, 15, 110]	Number	Entropy thông tin	IFS	accuracy, size, compuation time
2	[31, 53, 88]	Number	Entropy điều kiện	FS	accuracy, size, compuation time
3	[111]	Hybrid	Entropy kết hợp	Tập cổ điển	accuracy, size, compuation time
4	[112]	Number	Entropy bù	FS	accuracy, size, compuation time

1.5.2.3. độ đo khoảng cách

Độ đo khoảng cách là độ đo quan trong được sử dụng để đô lường độ khác biệt giữa hai phần tử hai tập hợp. Dựa trên tính chất đơn điệu của độ đo, một số độ đo được sử dụng để mở rộng cho bài toán rút gọn thuộc tính gồm có:

1. Khoảng cách Jacard [19]: Cho bảng quyết định $DT = (U, C, D, f)$. Với mọi $X, Y \subseteq U$, khoảng cách Jacard được xác định như sau:

$$D(X,Y) = 1 - \frac{|X \cap Y|}{|X \cup Y|} \tag{1.55}$$

2. Khoảng cách tri thức [48]: Cho bảng quyết định $DT = (U, C, D, f)$. Với mọi $P, Q \subseteq C$, với các tri thức tương ứng được kí hiệu bởi $K(P)$ và $K(Q)$. Trong đó $K(P) = \{[u]_P : u \in U\}$ và $K(Q) = \{[u]_Q : u \in U\}$. Khi đó, khoảng cách tri thức giữa P và Q theo tiếp cận Jacard được xác định như sau:

$$d_J(K(P), K(Q)) = 1 - \frac{1}{|U|^2} \sum_{u=1}^{|U|} \frac{\left|[u]_P \cap [u]_Q\right|}{\left|[u]_P \cup [u]_Q\right|} \tag{1.56}$$

Khi đó $\forall a \in C - B, B \subseteq C$, độ quan trọng của thuộc tính a với tập thuộc tính B được xác định như sau:

$$SIG_B(a) = d_J(K(B), K(B \cup D)) - d_J(K(B \cup \{a\}), K(B \cup \{a\} \cup D)) \tag{1.57}$$

Trên cơ sở đó, các phương pháp rút gọn thuộc tính theo tiếp cận độ đo khoảng cách được phát triển dựa trên mở rộng các độ đo này. Chi tiết các phương pháp được trình bày trong Bảng 1.7.

Bảng 1.7: *Tổng hợp phương pháp rút gọn thuộc tính theo khoảng cách*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Tiếp cận	Tập nền	Tiêu chuẩn đánh giá
1	[113, 33, 81, 114, 24]	Hybrid	KD	Tập cổ điển, FS, IFS	accuracy, size, compuation time
2	[29, 115, 116]	Number	GD	FS	accuracy, size, compuation time
3	[29]	Number	PD	FRS	accuracy, size, compuation time

1.5.3. Phương pháp rút gọn thuộc tính theo tiếp cận tôpô

Khái niệm không gian tôpô trên nền tập thô lần đầu tiên được đưa ra bởi Pawlack và công sự [14, 37, 18]. Trong đó, các phân hoạch được xem như là một cơ sở (subbase) của tôpô. Khi đó, Pawlack và các cộng sự cũng đã khẳng định cấu trúc đại số của tập thô với tôpô là tương đương trên nền tập rõ với các phép toán xấp xỉ dưới của tập thô tương đương với toán tử mền trong (INT) của tôpô và phép toán xấp xỉ trên của tập thô tương đương với toán tử miền ngoài (CLS) của tôpô [37, 18].

Bảng 1.8: *Tổng hợp phương pháp xây dựng tôpô theo tiếp cận tập thô*

STT	Tài liệu tham chiếu	Cơ sở tính toán
1	[18, 117, 38, 42, 20, 118, 40]	Không gian xấp xỉ
2	[39, 118, 117, 119, 120, 49, 48, 42, 40, 38]	Tập xấp xỉ trên và tập xấp xỉ dưới
3	[97, 57, 40, 46, 119, 121, 48, 122, 20, 20]	Không gian mẫu và quan hệ của các phép toán

Trên cơ các đề xuất ban đầu của Pawlack và các cộng sự, Lashin và các cộng sự đưa ra khái niệm biểu diễn tri thức trên không gian tôpô và phương pháp loại bỏ các tri thức dư thừa thông qua bảo toàn cấu trúc tôpô ban đầu [38]. Sau đó Zhu đề xuất khái niệm không gian tôpô trên họ các phủ của tập thô để rút gọn thuộc tính trong bảng quyết định không đầy đủ [123], tuy nhiên các nghiên cứu vẫn dừng lại ở giai đoạn đề xuất mà chưa có ứng dụng trên các bộ dữ liệu thực và đánh giá tính hiệu quả của mô hình đề xuất [124]. Gần đây có công trình nghiên cứu của Xie và các cộng sự có đề xuất hướng rút gọn thuộc tính dựa trên ma trận phân biệt sử dụng cấu trúc tôpô, trong đó tính chất phân biệt được mở rộng dựa trên cấu trúc tôpô thay vì tính chất của thuộc tính điều kiện hay thuộc tính quyết định [21, 94]. Shami và các cộng sự đề xuất xây dựng mô hình tập thô mới dựa trên không gian tôpô [45, 52]. Nhìn chung, mối quan hệ của không gian tôpô và mô hình tập thô đều dựa trên cấu trúc của các phân hoạch sinh bởi các quan hệ và các phép toán xấp xỉ của tập thô, xem Bảng 1.8. Do đó, về cơ bản hiện nay có ba loại phương pháp xây dựng không gian tôpô như sau:

Phương pháp sinh tôpô từ không gian xấp xỉ: Đối với mô hình tập thô truyền thống, không gian xấp xỉ là các phân hoạch hay tập các lớp tương đương [18, 117]. Do đó, các phân hoạch này luôn là các base của một tôpô. Khi đó để nghiên cứu cấu trúc tôpô, các nhà nghiên cứu thường phân tích cấu trúc của một base là đủ thông tin cần phân tích [38]. Do đó, khi một quan hệ xây dựng không gian xấp xỉ có tính chất khác nhau thì base cũng sẽ khác nhau. Đối với các phủ sinh bởi các quan hệ dung sai thì các phủ này được coi là các subbase, khi đó dựa trên các tính chất của base sinh từ subbase, có thể đề xuất các phương pháp xây dựng base từ các phủ này [42]. Trên nền FS và IFS, các nhà nghiên cứu cũng đề xuất phương pháp xây dựng tôpô từ các không gian xấp xỉ mờ [20] và xấp xỉ mờ trực cảm tương ứng [118, 40].

Phương pháp sinh tôpô từ cấu trúc tập thô: Mô hình tập thô có hai phép toán cơ bản là xấp xỉ trên và xấp xỉ dưới để đánh giá mức độ thô của một tập trong một không gian xấp xỉ. Pawlack và cộng sự [14, 18] cũng đã chỉ ra mối quan hệ giữa một tập với các tập xấp xỉ trên và dưới có quan hệ thứ tự, trong đó tập xấp xỉ trên luôn lớn hơn tập mục tiêu và tập mục tiêu luôn lớn tập xấp xỉ dưới. Do đó, kết hợp các tập xấp xỉ dưới và tập xấp xỉ trên luôn thỏa mãn là một cấu trúc tôpô [39, 118, 117, 119, 120, 49, 48, 42, 40, 38].

Phương pháp sinh tôpô dựa trên không gian mẫu: Dựa trên các phép toán của FRS và không gian mẫu của FS, các nhà nghiên cứu đề xuất các phương pháp khác nhau để xây dựng tôpô mờ [97, 57, 40, 46], trong đó các tác giả đưa ra nhiều cấu trúc tôpô mờ khác nhau dựa trên tính chất của quan hệ xậy dựng không gian xấp xỉ mờ và tính chất của phép toán xấp xỉ mờ trên, xấp xỉ mờ dưới tương ứng. Dựa trên các phép toán của IFRS và không gian mẫu IFS, các nhà nghiên cứu cũng mở rộng để xây dựng tôpô mờ trực cảm [119, 121, 48, 122, 20]. Hơn nữa, trong các công trình này, các tác giả còn chỉ rõ trường hợp nào hai tôpô bằng nhau và trường hợp nào hai tôpô bao thuộc lẫn nhau [122, 20]

Dựa trên khái niệm cơ sở β của không gian tôpô (U, τ). Lashin và các cộng sự [38] đã sử dụng khái niệm quan hệ dư thừa để định nghĩa tập rút gọn theo tiếp cận tôpô

như sau:

Định nghĩa 1.17 (Tập rút gọn theo tiếp cận tôpô [38])**.** Cho bảng quyết định $DT = (U, C, D, f)$, với τ_B và τC tương ứng là các tôpô của $B \subseteq C$ và C. Khi đó $r \in B$ được gọi là thuộc tính dư thừa trong B nếu: $\tau_B = \tau_{(B-\{r\})}$.

Khi đó: B được gọi là tập rút gọn của C khi và chỉ khi:

(i) $\tau_C = \tau_{(B)}$.

(ii) $\tau_C \neq \tau_{(B-\{\mathrm{r}\})}, \forall r \in C - B$.

Dựa trên các kết quả khảo sát về các phương pháp rút gọn thuộc tính theo tiếp cận tôpô và tập thô cho thấy các tiếp cận rút gọn thuộc tính trong bảng quyết định hiện nay còn gặp nhiều thách thức về *thời gian thực hiện* của thuật toán trên các tập dữ liệu kích thước lớn, khả năng cải thiện *kích thước* trên các bộ dữ liệu có số chiều lớn và khả năng cải thiện *nhiễu* trên các bộ dữ liệu xấu. Sau đây là các phân tích về nhược điểm của từng tiếp cận, từ đó đưa ra câu hỏi nghiên cứu và phương hướng giải quyết được thực hiện trong các Chương nghiên cứu tiếp theo luận án.

1) Các phương pháp rút gọn thuộc tính cho bảng quyết định số theo tiếp cận IFRS hiện nay chủ yếu dựa trên đề xuất các độ đo để xây dựng các tiêu chuẩn chọn lọc thuộc tính và định nghĩa tập rút gọn. Các độ đo này thường phụ thuộc rất lớn vào không gian xấp xỉ mờ trực cảm xây dựng được. Tuy nhiên, không gian xấp xỉ mờ trực cảm của các nghiên cứu hiện nay chưa phản ảnh rõ nét mối quan hệ của các đối tượng trong một tập, các giá trị về độ thuộc và độ không thuộc chưa được tính toán hiệu quả. Đây là nguyên nhân thứ nhất ảnh hưởng đến chất lượng của các thuộc tính chọn lọc được cho tập rút gọn. Nguyên nhân thứ hai là các độ đo theo tiếp cận miền dương mờ trực cảm và entropy thông tin mờ trực cảm của các nghiên cứu hiện nay đề xuất là các độ đo gián tiếp quan không gian xấp xỉ mờ trực cảm, do đó cũng ảnh hưởng một phần tới khả năng chọn lọc thuộc tính cho tập rút gọn. Cuối cùng là độ đo khoảng cách mờ hiện nay chưa được mở rộng cho không gian xấp xỉ mờ trực cảm.

2) Hầu hết các nghiên cứu về cấu trúc tôpô rút gọn theo tiếp cận tập thô hiện nay còn chưa đầy đủ, chưa rõ ràng về nền tảng lý thuyết, dẫn đến các phương pháp rút gọn

thuộc tính cho các tập dữ liệu thực hiện nay còn rất hạn chế về số lượng cũng như chất lượng mô hình lý thuyết. Do đó, cần phải phát triển khung nền tảng lý thuyết tôpô cho bài toán rút gọn thuộc tính. Trong đó cần phải chỉ rõ, phương pháp sinh tôpô theo tiếp cận nào là hiệu quả cho bài toán rút gọn thuộc tính và các phép toán đại số trên tôpô cần được mở rộng để phát triển các phương pháp rút gọn thuộc tính hiệu quả trên các bộ dữ liệu có số chiều lớn.

1.6. Kết luận Chương 1

Chương 1 đã giới thiệu khái quát về bài toán rút gọn thuộc tính và phân loại phương pháp rút gọn thuộc tính. Trình bày các cơ sở lý thuyết quan trọng để thực hiện trong các Chương nghiên cứu tiếp theo của luận án như sau:

- Trình bày khái quát các khái niệm cơ bản về mô hình lý thuyết tập thô truyền thống và tập mờ trực cảm. Ý nghĩa và vai trò của tập mờ trực cảm trong việc cải thiện nhiễu, các độ đo cơ bản là cơ sở kiến thức quan trọng được sử dụng trong Chương 2 của luận án.

- Trình bày khái quát các khái niệm cơ bản về không gian tôpô, các nghiên cứu liên quan đến phương pháp xây dựng tôpô trên không gian xấp xỉ mờ là cơ sở kiến thức quan trọng được sử dụng trong Chương 3 của luận án.

- Trình bày khái quát các nghiên cứu liên quan đến phương pháp xây dựng tôpô theo tiếp cận tập thô, các phương pháp xây dựng các phép toán cho cấu trúc tôpô đại số là các kiến thức quan trọng được sử dụng trong Chương 4 của luận án.

CHƯƠNG 2. PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRONG BẢNG QUYẾT ĐỊNH THEO TIẾP CẬN TẬP THÔ MỜ TRỰC CẢM

2.1. Mở đầu

Để rút gọn thuộc tính trực tiếp trên các bảng quyết định miền giá trị số liên tục. Khoảng hơn mười năm trở lại đây, các nhà nghiên cứu thường mở rộng mô hình tập thô truyền thống sang mô hình tập thô lân cận [75, 125, 83, 84, 77], mô hình tập thô mờ [96, 55, 58, 33, 126]. Bên cạnh đó, mô hình tập thô xác suất [29, 127, 128], mô hình tập thô biến thiên về độ chính xác [129, 89] cũng được các nhà nghiên cứu mở rộng cho các trường hợp dữ liệu nhiễu. Tuy nhiên khả năng cải thiện nhiễu của các tiếp cận trên vẫn còn chưa hiệu quả. Các nghiên cứu gần đây đã cho thấy tập nền mờ trực cảm có khả năng biểu diễn khá chính xác mối quan hệ của các đối tượng, nhất là trên các môi trường dữ liệu không chắc chắn. Do đó, mô hình IFRS được nhiều nhà nghiên cứu quan tâm và phát triển [34, 93, 27].

Gần đây phương pháp rút gọn thuộc tính theo tiếp cận IFRS do Tan và các cộng sự đề xuất [36, 130] cho tập rút gọn hiệu quả về độ chính xác phân lớp. Tuy nhiên khả năng cải thiện độ chính xác phân lớp trên các bộ dữ liệu nhiễu vẫn còn hạn chế. Nguyên nhân thứ nhất, không gian xấp xỉ mờ trực cảm của các nghiên cứu này chưa phản ảnh rõ nét mối quan hệ của các đối tượng trong một tập, các giá trị về độ thuộc và độ không thuộc chưa được tính toán hiệu quả. Nguyên nhân thứ hai, các độ đo theo tiếp cận miền dương mờ trực cảm và entropy thông tin mờ trực cảm là các độ đo gián tiếp qua không gian xấp xỉ mờ trực cảm, do đó cũng ảnh hưởng một phần tới khả năng chọn lọc thuộc tính cho tập rút gọn. Do đó, chương này trình bày phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm với các bước chính như sau:

- Mở rộng độ đo khoảng cách mờ trên nền tập mờ trực cảm và xây dựng độ đo khoảng cách giữa các phân hoạch mờ trực cảm được gọi tắt là *khoảng cách mờ trực cảm*.

- Xây dựng độ đo đánh giá độ quan trọng của thuộc tính dựa trên khoảng cách giữa các phân hoạch mờ trực cảm, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm δ-equal để định nghĩa tập rút gọn và xây dựng điều kiện dừng của thuật toán

Trên cơ sở đó, luận án đề xuất phương pháp lai ghép filter - wrapper tìm tập rút gọn hiệu quả về độ chính xác phân lớp trên các tập dữ liệu nhiễu. Ngoài ứng dụng của độ đo đề xuất cho bài toán rút gọn thuộc tính, độ đo này có thể áp dụng cho một số bài toán phân lớp, dự báo, hỗ trợ ra quyết định có liên quan đến kĩ thuật tính toán mềm trên tập các số mờ trực cảm.

Các kết quả nghiên cứu đã được công bố trên các công trình nghiên cứu [CT3, CT4].

2.2. Xây dựng độ đo khoảng cách mờ trực cảm

2.2.1. Khoảng cách giữa hai tập mờ trực cảm

Vì mỗi phần tử của tập mờ trực cảm được đặc trưng bởi độ thành viên và độ không thành viên. Do đó, trước khi xây dựng độ đo khoảng cách giữa hai tập mờ trực cảm, luận án xây dựng bổ đề về số mờ trực cảm, làm cơ sở để xây dựng các độ đo tiếp theo của luận án.

Bổ đề 2.1 [Số mờ trực cảm]**.** *Cho ba số thực* $a,b,c \in [0,1]$. *Khi đó:*

1) Nếu $a \geq b$ *thì* $a-b \geq \min(a,\ c) - \min(b,\ c)$

2) Nếu $a \leq b$ *thì* $a-b \leq max(a,\ c) - max(b,\ c)$

Mệnh đề 2.1 (Quan hệ của các IFS)**.** *Cho* $\tilde{\tilde{X}}, \tilde{\tilde{Y}}, \tilde{\tilde{Z}}$ *là các tập mờ trực cảm xác định trên* U, *với* U *là tập không rỗng các đối tượng. Khi đó:*

1) Nếu $\tilde{\tilde{X}} \subseteq \tilde{\tilde{Y}}$ *thì* $\left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right| \geq \left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right|$

2) Nếu $\tilde{\tilde{X}} \subseteq \tilde{\tilde{Y}}$ *thì* $\left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

3) $\left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right| + \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

Chứng minh. Sau đây ta lần lượt chứng minh từng tính chất như sau:

1) Vì $\tilde{\tilde{X}} \subseteq \tilde{\tilde{Y}}$, do đó với mọi $u \in U$ ta có $\mu_{\tilde{\tilde{Y}}}(u) \geq \mu_{\tilde{\tilde{X}}}(u)$ và $\nu_{\tilde{\tilde{Y}}}(u) \leq \nu_{\tilde{\tilde{X}}}(u)$. Áp dụng bổ đề 2.1 ta có:

(1): $\mu_{\tilde{\tilde{Y}}}(u) - \mu_{\tilde{\tilde{X}}}(u) \geq \min\left(\mu_{\tilde{\tilde{Y}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right) - \min\left(\mu_{\tilde{\tilde{X}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \sum_{i=1}^{|U|} \mu_{\tilde{\tilde{Y}}}(i) - \sum_{i=1}^{|U|} \mu_{\tilde{\tilde{X}}}(i) \geq \sum_{i=1}^{|U|} \min\left(\mu_{\tilde{\tilde{Y}}}(i), \mu_{\tilde{\tilde{Z}}}(i)\right) - \sum_{i=1}^{|U|} \min\left(\mu_{\tilde{\tilde{X}}}(i), \mu_{\tilde{\tilde{Z}}}(i)\right)$

(2): $\nu_{\tilde{\tilde{Y}}}(u) - \nu_X(u) \leq max\left(\nu_{\tilde{\tilde{Y}}}(u), \nu_{\tilde{\tilde{Z}}}(u)\right) - max\left(\nu_X(u), \nu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \sum_{i=1}^{|U|} \nu_{\tilde{\tilde{Y}}}(i) - \sum_{i=1}^{|U|} \nu_X(i) \leq \sum_{i=1}^{|U|} max\left(\nu_{\tilde{\tilde{Y}}}(i), \nu_{\tilde{\tilde{Z}}}(i)\right) - \sum_{i=1}^{|U|} max\left(\nu_{\tilde{\tilde{X}}}(i), \nu_{\tilde{\tilde{Z}}}(i)\right)$

Từ (1) và (2) ta có: $\left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right| \Leftrightarrow \left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right| \geq \left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right|$

2) Vì $\tilde{\tilde{X}} \subseteq \tilde{\tilde{Y}}$, do đó với mọi $u \in U$ ta có $\mu_{\tilde{\tilde{Y}}}(u) \geq \mu_{\tilde{\tilde{X}}}(u)$ và $\nu_{\tilde{\tilde{Y}}}(u) \leq \nu_{\tilde{\tilde{X}}}(u)$. Áp dụng bổ đề 2.1 ta có:

(3): $\mu_{\tilde{\tilde{Y}}}(u) \geq \mu_{\tilde{\tilde{X}}}(u) \Leftrightarrow \min\left(\mu_{\tilde{\tilde{Y}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right) \geq \min\left(\mu_{\tilde{\tilde{X}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \mu_{\tilde{\tilde{Z}}}(u) - \min\left(\mu_{\tilde{\tilde{X}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right) \geq \mu_{\tilde{\tilde{Z}}}(u) - \min\left(\mu_{\tilde{\tilde{Y}}}(u), \mu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \sum_{i=1}^{|U|} \mu_{\tilde{\tilde{Z}}}(i) - \sum_{i=1}^{|U|} \min\left(\mu_{\tilde{\tilde{X}}}(i), \mu_{\tilde{\tilde{Z}}}(i)\right) \geq \sum_{i=1}^{|U|} \mu_{\tilde{\tilde{Z}}}(i) - \sum_{i=1}^{|U|} \min\left(\mu_{\tilde{\tilde{Y}}}(i), \mu_{\tilde{\tilde{Z}}}(i)\right)$

(4): $\nu_{\tilde{\tilde{Y}}}(u) \leq \nu_{\tilde{\tilde{X}}}(u) \Leftrightarrow max\left(\nu_{\tilde{\tilde{Y}}}(u), \nu_{\tilde{\tilde{Z}}}(u)\right) \leq max\left(\nu_{\tilde{\tilde{X}}}(u), \nu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \nu_{\tilde{\tilde{Z}}}(u) - max\left(\nu_{\tilde{\tilde{X}}}(u), \nu_{\tilde{\tilde{Z}}}(u)\right) \leq \nu_{\tilde{\tilde{Z}}}(u) - max\left(\nu_{\tilde{\tilde{Y}}}(u), \nu_{\tilde{\tilde{Z}}}(u)\right)$

$\Leftrightarrow \sum_{i=1}^{|U|} \nu_{\tilde{\tilde{Z}}}(i) - \sum_{i=1}^{|U|} max\left(\nu_{\tilde{\tilde{X}}}(i), \nu_{\tilde{\tilde{Z}}}(i)\right) \leq \sum_{i=1}^{|U|} \nu_{\tilde{\tilde{Z}}}(i) - \sum_{i=1}^{|U|} max\left(\nu_{\tilde{\tilde{Y}}}(i), \nu_{\tilde{\tilde{Z}}}(i)\right)$

Từ (3) và (4) ta có: $\left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

3) Vì $\tilde{\tilde{X}} \cap \tilde{\tilde{Z}} \subseteq \tilde{\tilde{X}}$ và $\tilde{\tilde{X}} \cap \tilde{\tilde{Y}} \subseteq \tilde{\tilde{Y}}$. Từ 1) và 2) ta có:

(5): $\left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right| \geq \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

(6): $\left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

Từ (5) và (6) ta có: $\left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right| + \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$. đpcm. □

Mệnh đề 2.2 (Khoảng cách giữa hai IFS)**.** *Cho hai tập mờ trực cảm $\tilde{\tilde{X}}, \tilde{\tilde{Y}}$ xác định trên U, với U là tập không rỗng các đối tượng. Khi đó $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right) = \left|\tilde{\tilde{X}} \cup \tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right|$ là khoảng cách giữa hai tập mờ trực cảm $\tilde{\tilde{X}}, \tilde{\tilde{Y}}$.*

Chứng minh. Vì $\left|\tilde{\tilde{X}} \cup \tilde{\tilde{Y}}\right| \geq \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right|$ do đó $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right) \geq 0$. Để chứng minh $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right)$ là độ đo khoảng cách, ta cần chứng minh $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right)$ thỏa mãn bất đẳng thức tam giác. Thật vậy ta có:

(7): $\left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right| + \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Z}} \cap \tilde{\tilde{Y}}\right|$

(8): $\left|\tilde{\tilde{X}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right| + \left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{X}}\right| \geq \left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right|$

Từ (7) và (8) ta có:

(9): $\left(\left|\tilde{\tilde{X}}\right| + \left|\tilde{\tilde{Y}}\right| - 2\left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right|\right) + \left(\left|\tilde{\tilde{X}}\right| + \left|\tilde{\tilde{Z}}\right| - 2\left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right|\right) \geq \left|\tilde{\tilde{Y}}\right| + \left|\tilde{\tilde{Z}}\right| - 2\left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right|$.

Hơn nữa với mọi $x, y \in \mathbb{R}$ ta luôn có $\max(x,y) = x + y - \min(x,y)$ và $\min(x,y) = x + y - \max(x,y)$ do đó với mọi $u \in U$:

(10): $\max\left(\mu_X(u), \mu_Y(u)\right) = \mu_X(u) + \mu_Y(u) - \min\left(\mu_X(u), \mu_Y(u)\right)$

(11): $\min\left(\nu_X(u), \nu_Y(u)\right) = \nu_X(u) + \nu_Y(u) - \max\left(\nu_X(u), \nu_Y(u)\right)$

Từ (10) và (11) ta có:

(12): $\left|\tilde{\tilde{X}} \cup \tilde{\tilde{Y}}\right| = \left|\tilde{\tilde{X}}\right| + \left|\tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right|$.

Từ (9) và (12) ta có: $\left(\left|\tilde{\tilde{X}} \cup \tilde{\tilde{Y}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Y}}\right|\right) + \left(\left|\tilde{\tilde{X}} \cup \tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{X}} \cap \tilde{\tilde{Z}}\right|\right) \geq \left|\tilde{\tilde{Y}} \cup \tilde{\tilde{Z}}\right| - \left|\tilde{\tilde{Y}} \cap \tilde{\tilde{Z}}\right|$ hay $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right) + \tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Z}}\right) \geq \tilde{\tilde{d}}\left(\tilde{\tilde{Y}}, \tilde{\tilde{Z}}\right)$. Do đó $\tilde{\tilde{d}}\left(\tilde{\tilde{X}}, \tilde{\tilde{Y}}\right)$ là một độ đo khoảng cách. □

2.2.2. Khoảng cách giữa hai phân hoạch mờ trực cảm

Mệnh đề 2.3 (Độ đo khoảng cách mờ trực cảm)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và hai phân hoạch $[\tilde{\tilde{X}}]$, $[\tilde{\tilde{Y}}]$ tương ứng của $X, Y \subseteq C$. Khi đó $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}], [\tilde{\tilde{Y}}]\right)$ là khoảng cách giữa hai phân hoạch mờ trực cảm.*

Chứng minh. Thật vậy, ta luôn có $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}], [\tilde{\tilde{Y}}]\right) \geq 0$ và $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}], [\tilde{\tilde{Y}}]\right) = \tilde{\tilde{d}}\left([\tilde{\tilde{X}}], [\tilde{\tilde{Y}}]\right)$.

Khi đó để $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)$ là một khoảng cách, ta cần chứng minh bất đẳng thức tam giác $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)+\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Z}}]\right)\geq\tilde{\tilde{d}}\left([\tilde{\tilde{Y}}],[\tilde{\tilde{Z}}]\right)$.

Từ mệnh đề 2.2, với mọi $u\in U$ ta có $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)+\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Z}}]\right)\geq\tilde{\tilde{d}}\left([\tilde{\tilde{Y}}],[\tilde{\tilde{Z}}]\right)$. Khi đó: $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)+\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Z}}]\right)$

$=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\cup[i]_{\tilde{\tilde{Y}}}\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{Y}}}\right|\right)$

$+\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\cup[i]_{\tilde{\tilde{Z}}}\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{Z}}}\right|\right)$

$=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\tilde{\tilde{d}}\left([i]_{\tilde{\tilde{X}}},[i]_{\tilde{\tilde{Y}}}\right)+\frac{1}{|U|^2}\sum_{i=1}^{|U|}\tilde{\tilde{d}}\left([i]_{\tilde{\tilde{X}}},[i]_{\tilde{\tilde{Z}}}\right)\geq\frac{1}{|U|^2}\sum_{i=1}^{|U|}\tilde{\tilde{d}}\left([i]_{\tilde{\tilde{Y}}},[i]_{\tilde{\tilde{Z}}}\right)$

$=\tilde{\tilde{d}}\left([\tilde{\tilde{Y}}],[\tilde{\tilde{Z}}]\right)$. Ta có đpcm. □

Định nghĩa 2.1 (Khoảng cách giữa hai phân hoạch mờ trực cảm)**.** Cho bảng quyết định $DT=(U,C,D,f)$ và hai phân hoạch $[\tilde{\tilde{X}}]$, $[\tilde{\tilde{Y}}]$ tương ứng của $X,Y\subseteq C$. Khi đó khoảng cách giữa $[\tilde{\tilde{X}}]$, $[\tilde{\tilde{Y}}]$ được xác định bởi:

$$\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\cup[i]_{\tilde{\tilde{Y}}}\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{Y}}}\right|\right)\tag{2.1}$$

Mệnh đề 2.4 (Độ đo phân hạt)**.** *Cho bảng quyết định $DT=(U,C,D,f)$ và $[\tilde{\tilde{X}}]$, $[\tilde{\tilde{Y}}]$ tương ứng là các phân hoạch của X và $Y=X\cup D$ với $X\subseteq C$. Khi đó $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[\tilde{\tilde{Y}}]\right)$ là một độ đo khoảng cách.*

Chứng minh. Từ mệnh đề 2.3 ta có: $\tilde{\tilde{d}}\left([\tilde{\tilde{X}}],[X\tilde{\tilde{\cup}}D]\right)$

$=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\cup[i]_{[X\cup D]}\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{[X\cup D]}\right|\right)$

$=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\cup\left([i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{D}}}\right)\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{D}}}\right|\right)$

$=\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{\tilde{\tilde{X}}}\right|-\left|[i]_{\tilde{\tilde{X}}}\cap[i]_{\tilde{\tilde{D}}}\right|\right)$ □

Định nghĩa 2.2 (Khoảng cách phân hạt mờ trực cảm)**.** Cho bảng quyết định $DT=$

(U,C,D,f) và $[\overset{\approx}{X}]$, $[\overset{\approx}{Y}]$ tương ứng là các phân hoạch của X và $Y = X \cup D$ với $X \subseteq C$. Khi đó khoảng cách giữa $[\overset{\approx}{X}]$, $[X \cup \overset{\approx}{D}]$ được xác định bởi:

$$\overset{\approx}{d}\left([\overset{\approx}{X}],[\overset{\approx}{Y}]\right) = \frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|[i]_{[\overset{\approx}{X}]}\right| - \left|[i]_{[\overset{\approx}{X}]} \cap [i]_{[\overset{\approx}{D}]}\right|\right) \tag{2.2}$$

Mệnh đề 2.5 (Tính chất phản đơn điệu của độ đo phân hạt)**.** *Cho bảng quyết định $DT = (U,C,D,f)$ và $[\overset{\approx}{X}]$, $[\overset{\approx}{Y}]$ tương ứng là các phân hoạch của X và Y với $X \subseteq Y \subseteq C$. Khi đó $\overset{\approx}{d}\left([\overset{\approx}{X}],[X \cup \overset{\approx}{D}]\right) \geq \overset{\approx}{d}\left([\overset{\approx}{Y}],[Y \cup \overset{\approx}{D}]\right)$ với mọi $u \in U$.*

Chứng minh. Vì $X \subseteq Y$, do đó $[\overset{\approx}{Y}] \leq [\overset{\approx}{X}]$ nghĩa là $[u]_{[\overset{\approx}{Y}]} \subseteq [u]_{[\overset{\approx}{X}]}$ với $1 \leq u \leq |U|$. Khi đó $\left|[u]_{[\overset{\approx}{Y}]}\right| \leq \left|[u]_{[\overset{\approx}{X}]}\right|$. Với mọi $u \in U$ ta có:

(13): $\left|[u]_{[\overset{\approx}{Y}]}\right| - \left|[u]_{[\overset{\approx}{Y}]} \cap [u]_{[\overset{\approx}{D}]}\right| = \sum_{k=1}^{|U|}\mu_{[u]_{[\overset{\approx}{Y}]}}(k) - \sum_{k=1}^{|U|}\min\left\{\mu_{[u]_{[\overset{\approx}{Y}]}}(k), \mu_{[u]_{[\overset{\approx}{D}]}}(k)\right\}$

(14): $\left|[u]_{[\overset{\approx}{X}]}\right| - \left|[u]_{[\overset{\approx}{X}]} \cap [u]_{[\overset{\approx}{D}]}\right| = \sum_{k=1}^{|U|}\mu_{[u]_{[\overset{\approx}{X}]}}(k) - \sum_{k=1}^{|U|}\min\left\{\mu_{[u]_{[\overset{\approx}{X}]}}(k), \mu_{[u]_{[\overset{\approx}{D}]}}(k)\right\}$

Với mọi $k \in [u]_{[\overset{\approx}{D}]}$, nếu $\mu_{[u]_{[\overset{\approx}{D}]}}(k) = 1$ thì

(15): $\left|[u]_{[\overset{\approx}{X}]}\right| - \left|[u]_{[\overset{\approx}{X}]} \cap [u]_{[\overset{\approx}{D}]}\right| = 0 = \left|[u]_{[\overset{\approx}{Y}]}\right| - \left|[u]_{[\overset{\approx}{Y}]} \cap [u]_{[\overset{\approx}{D}]}\right|$

Với mọi $k \notin [u]_{[\overset{\approx}{D}]}$ nếu $\mu_{[u]_{[\overset{\approx}{D}]}}(k) = 0$ thì

(16): $\left|[u]_{[\overset{\approx}{X}]}\right| - \left|[u]_{[\overset{\approx}{X}]} \cap [u]_{[\overset{\approx}{D}]}\right| \geq \left|[u]_{[\overset{\approx}{Y}]}\right| - \left|[u]_{[\overset{\approx}{Y}]} \cap [u]_{[\overset{\approx}{D}]}\right| \Leftrightarrow \left|[u]_{[\overset{\approx}{X}]}\right| \geq \left|[u]_{[\overset{\approx}{Y}]}\right|$

Từ (15) và (16) ta có:

$$\left|[u]_{[\overset{\approx}{X}]}\right| - \left|[u]_{[\overset{\approx}{X}]} \cap [u]_{[\overset{\approx}{D}]}\right| \geq \left|[u]_{[\overset{\approx}{Y}]}\right| - \left|[u]_{[\overset{\approx}{Y}]} \cap [u]_{[\overset{\approx}{D}]}\right|$$

$$\Leftrightarrow \frac{1}{|U|^2}\sum_{u=1}^{|U|}\left(\left|[u]_{[\overset{\approx}{X}]}\right| - \left|[u]_{[\overset{\approx}{X}]} \cap [u]_{[\overset{\approx}{D}]}\right|\right) \geq \frac{1}{|U|^2}\sum_{u=1}^{|U|}\left(\left|[u]_{[\overset{\approx}{Y}]}\right| - \left|[u]_{[\overset{\approx}{Y}]} \cap [u]_{[\overset{\approx}{D}]}\right|\right)$$

$$\Leftrightarrow \overset{\approx}{d}\left([\overset{\approx}{X}],[X \cup \overset{\approx}{D}]\right) \geq \overset{\approx}{d}\left([\overset{\approx}{Y}],[Y \cup \overset{\approx}{D}]\right).$$ đpcm □

2.3. Rút gọn thuộc tính trong bảng quyết định sử dụng độ đo khoảng cách mờ trực cảm

2.3.1. Đề xuất thuật toán tìm tập rút gọn theo phương pháp lai ghép filter - wrapper, sử dụng độ đo khoảng cách mờ trực cảm

Định nghĩa 2.3 (Ma trận δ equal)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai ma trận quan hệ mờ trực cảm $\tilde{\tilde{M}}_B = [b_{ij}]_{n\times n}$, $\tilde{\tilde{M}}_C = [c_{ij}]_{n\times n}$ xác định trên B và C với $B \subseteq C$, $n = |U|$. Khi đó $\tilde{\tilde{M}}_B$ và $\tilde{\tilde{M}}_C$ được gọi là δ - equal khi và chỉ khi:

1) $\sup_{i,j=1}^{n} \left|\mu(b_{ij}) - \mu(c_{ij})\right| \leq 1 - \delta$

2) $\sup_{i,j=1}^{n} \left|\nu(b_{ij}) - \nu(c_{ij})\right| \leq 1 - \delta$

Trong đó $\sup_{i,j=1}^{n}$ cho biết sự khác biệt lớn nhất của hai ma trận quan hệ mờ trực cảm đạt được tại vị trí i, j, với $\delta \in [0,1]$. Ta kí hiệu $\tilde{\tilde{M}}_B \overset{\delta}{=} \tilde{\tilde{M}}_C$.

Định nghĩa 2.4 (Độ quan trọng của thuộc tính)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và tập thuộc tính $B \subseteq C$. Khi đó độ quan trọng của thuộc tính $a \in C - B$ với tập thuộc tính B được định nghĩa bởi công thức sau:

$$SIG_B(a) = \tilde{\tilde{d}}\left([\tilde{\tilde{B}}], [B \tilde{\tilde{\cup}} D]\right) - \tilde{\tilde{d}}\left([B \tilde{\tilde{\cup}} \{a\}], [B \cup \{\tilde{\tilde{a}}\} \cup D]\right) \tag{2.3}$$

Định nghĩa 2.5 (Tập rút gọn)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và tập thuộc tính $B \subseteq C$. Khi đó tập thuộc tính B được gọi là tập rút gọn nếu:

1) $[B \tilde{\tilde{\cup}} D] \overset{\delta}{=} [C \tilde{\tilde{\cup}} D]$;

2) $\forall b \in B,\ [B - \{\tilde{\tilde{b}}\} \cup D] \overset{\delta}{\neq} [C \tilde{\tilde{\cup}} D]$.

Dựa theo các thành phần cơ bản của mô hình rút gọn thuộc tính đề xuất đã được định nghĩa bên trên, sau đây luận án đề xuất thuật toán rút gọn thuộc tính theo phương pháp lai ghép filter-wrapper hai bước.

Thuật toán đề xuất bao gồm có hai giai đoạn, giai đoạn filter - W_δ và giai đoạn W_A. Trong đó, bước filter sử dụng định nghĩa 2.4 và định nghĩa 2.5 để xác định tập rút gọn ứng viên mức δ. Kết hợp với với mô hình phân lớp $Model$, bước W_δ để xác định tập

Thuật toán 2.1 Thuật toán filter - wrapper hai giai đoạn sử dụng khoảng cách mờ trực cảm (IFD)

Input: $DT = (U, C, D, f)$, mô hình phân lớp $Model$, $\Delta = \{0.1, 0.2, ..., 0.9\}$
Output: Tập rút gọn R

1: $R_W^A \leftarrow \emptyset$;
2: $R_W^\delta \leftarrow \emptyset$;
3: **for all** $c \in C$ **do**
4: computation $[c]^{\approx}$;
5: **end for**
6: **for all** $\delta \in \Delta$ **do**
7: $R_F^\delta \leftarrow \emptyset$;
8: **while** $[R_F^\delta \cup D]^{\approx} \neq^{\delta} [C \cup D]^{\approx}$ **do**
9: $c_m \in C - R_F^\delta | SIG_{R_F^\delta}(c_m) = \underset{c \in C - R_F^\delta}{Max} \left\{ SIG_{R_F^\delta}(c) \right\}$; {Giai đoạn filter}
10: $R_F^\delta := R_F^\delta \cup \{c_m\}$;
11: **end while**
12: **if** $ACC(Model, R_F^\delta) > ACC(Model, R_W^\delta)$ **then**
13: $R_W^\delta = R_F^\delta$; {Giai đoạn wrapper delta (W_δ)}
14: **end if**
15: **end for**
16: **for** $(i = 1; i < |R_W^\delta|; i++)$ **do**
17: **if** $ACC(Model, R_W^\delta[0:i]) > ACC(Model, R_W^A)$ **then**
18: $R_W^A = R_W^\delta[0:i]$; {Giai đoạn wrapper attribute (W_A)}
19: **end if**
20: **end for**
21: **return** R_W^A;

rút gọn ứng viên tốt nhất trong toàn bộ các giá trị δ. Kết thúc giai đoạn filter - W_δ, chuyển sang giai đoạn W_A. Giai đoạn W_A truy vết tập con nào của tập rút gọn ứng viên mức δ có độ chính xác phân lớp cao nhất với mô hình phân lớp $Model$. Kết thúc giai đoạn W_A ta thu được tập rút gọn thực sự của thuật toán. Sau đây là các bước chi tiết của thuật toán đề xuất.

Trong đó: R_W^A là kí hiệu cho tập rút gọn thu được, R_F^δ là tập thuộc tính lọc được tại mức δ, R_W^δ là tập rút gọn ứng viên có độ chính xác phân lớp cao nhất tại mức δ. ACC là hàm đánh giá độ chính xác phân lớp của tập rút gọn cho trước trên mô hình phân

lớp *Model*. Mỗi giá trị δ có bước nhảy là 0.1. Sau đây là phần đánh giá độ phức tạp của thuật toán đề xuất.

Trước tiên, luận án kí hiệu $|U|$ là số các đối tượng và $|C|$ là số các thuộc tính trong bảng quyết định $DT = (U, C, D, f)$, $|R^{\delta}|$ là kích thước của tập thuộc tính rút gọn ứng viên, $|\delta|$ là số lượng các giá trị δ cần xét. Gọi $\mathbb{T}$ là thời gian thực hiện của mô hình phân lớp *Model*. Khi đó, độ phức tạp của thuật toán được trình bày trong Bảng 2.1 như sau:

Bảng 2.1: *Độ phức tạp của thuật toán IFD*

Dòng lệnh	Độ phức tạp
Dòng 3 - 5	$O\left(\lvert C\rvert \lvert U\rvert^2\right)$
Dòng 8 - 11	$O(\lvert C\rvert^2 \lvert U\rvert^2)$
Dòng 13	$O(\mathbb{T})$
Dòng 6-14	$O(\mathbb{T}\lvert\Delta\rvert\lvert C\rvert^2\lvert U\rvert^2)$
Dòng 16	$O(\mathbb{T}\lvert R_W^{\delta}\rvert)$
Tổng	$O\left(\lvert C\rvert \lvert U\rvert^2\right) + O(\mathbb{T}\lvert\Delta\rvert\lvert C\rvert^2\lvert U\rvert^2) + O(\mathbb{T}\lvert R_W^{\delta}\rvert)$

Để minh họa quá trình hoạt động của thuật toán đề xuất, sau đây là phần trình bày ví dụ số minh họa trên bảng quyết định số như sau. Cho bảng quyết định $DT = (U, C, D, f)$ được trình bày như trong bảng 1.3. Các bước của thuật toán được thực hiện tuần tự như sau:

Giai đoạn khởi tạo:

- Đặt: $R_W^A \leftarrow \emptyset$; $R_W^{\delta} \leftarrow \emptyset$;

- Tính các phân hoạch $[\overset{\approx}{a}]$, $[\overset{\approx}{b}]$, $[\overset{\approx}{c}]$, $[\overset{\approx}{d}]$, $[\overset{\approx}{e}]$, $[\overset{\approx}{f}]$, $[\overset{\approx}{D}]$, $[\overset{\approx}{C}]$, $[\overset{\approx}{B}]$. Trong đó: $[\overset{\approx}{B}]$ là ma trận quan hệ mờ trực cảm thô nhất.

Giai đoạn Filter - W_{δ}

Lần lượt thử các giá trị δ từ $0.1 \rightarrow 0.9$ với mỗi bước nhảy là 0.1. Khi đó giá trị δ nào cho tập rút gọn ứng viên có độ chính xác cao nhất, ta sẽ chọn tập rút gọn ứng viên đó cho giai đoạn W_A của thuật toán. Giả sử tập rút gọn ứng viên có độ chính xác cao nhất tại giá trị $\delta = 0.8$, sau đây là chi tiết các bước thực hiện tại giá trị $\delta = 0.8$.

$R_F^\delta = \emptyset$

Vì $[R_F^\delta \overset{\approx}{\cup} D] \overset{\delta}{\neq} [C \overset{\approx}{\cup} D]$, do đó:

- Tính:

$SIG_{R_F^\delta}(a) = \overset{\approx}{d}\left(R_F^\delta, [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{a\}], [R_F^\delta \cup \overset{\approx}{\{a\}} \cup D]\right) = 0.5 - 0.31 = 0.19$

$SIG_{R_F^\delta}(b) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{b\}], [R_F^\delta \cup \overset{\approx}{\{b\}} \cup D]\right) = 0.5 - 0.43 = 0.07$

$SIG_{R_F^\delta}(c) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{c\}], [R_F^\delta \cup \overset{\approx}{\{c\}} \cup D]\right) = 0.5 - 0.36 = 0.14$

$SIG_{R_F^\delta}(d) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left(P^{\approx}_{R_F^\delta \cup \{d\}}, [R_F^\delta \cup \overset{\approx}{\{d\}} \cup D]\right) = 0.5 - 0.38 = 0.12$

$SIG_{R_F^\delta}(e) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{e\}], [R_F^\delta \cup \overset{\approx}{\{e\}} \cup D]\right) = 0.5 - 0.26 = 0.24$

$SIG_{R_F^\delta}(f) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{f\}], [R_F^\delta \cup \overset{\approx}{\{f\}} \cup D]\right) = 0.5 - 0.26 = 0.24$

- Chọn e vì $SIG_{R_F^\delta}(e)$ lớn nhất, do đó $R_F^\delta = \{e\}$.

Vì $\min\left(1 - \sup_{i,j=1}^{6} \left|\mu(b_{ij}) - \mu(c_{ij})\right|, 1 - \sup_{i,j=1}^{6} \left|\nu(b_{ij}) - \nu(c_{ij})\right|\right) = 0.4 < \delta$ hay $[\{R_F^\delta \overset{\approx}{-} b\} \cup D] \neq (\delta)[C \overset{\approx}{\cup} D]$, do đó tiếp tục vòng lặp ta có:

- Tính:

$SIG_{R_F^\delta}(a) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{a\}], [R_F^\delta \cup \overset{\approx}{\{a\}} \cup D]\right) = 0.26 - 0.23 = 0.03$

$SIG_{R_F^\delta}(b) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{b\}], [R_F^\delta \cup \overset{\approx}{\{b\}} \cup D]\right) = 0.26 - 0.25 = 0.01$

$SIG_{R_F^\delta}(c) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left([R_F^\delta \overset{\approx}{\cup} \{c\}], [R_F^\delta \cup \overset{\approx}{\{c\}} \cup D]\right) = 0.26 - 0.18 = 0.08$

$SIG_{R_F^\delta}(d) = \overset{\approx}{d}\left([R_F^\delta], [R_F^\delta \overset{\approx}{\cup} D]\right) - \overset{\approx}{d}\left(P^{\approx}_{R_F^\delta \cup \{d\}}, [R_F^\delta \cup \overset{\approx}{d} \cup D]\right) = 0.26 - 0.20 = 0.06$

$SIG_{R_F^\delta}(f) = \tilde{\tilde{d}}\left([R_F^\delta], [R_F^\delta \tilde{\cup} D]\right) - \tilde{\tilde{d}}\left([R_F^\delta \tilde{\cup} \{f\}], [R_F^\delta \cup \tilde{\{f\}} \cup D]\right) = 0.26 - 0.26 = 0.00$

- Chọn c vì $SIG_{R_F^\delta}(c)$ lớn nhất, do đó $R_F^\delta = R_F^\delta \cup \{c\} = \{e,c\}$.

Vì $\min\left(1 - \sup_{i,j=1}^{6} \left|\mu(b_{ij}) - \mu(c_{ij})\right|, 1 - \sup_{i,j=1}^{6} \left|\nu(b_{ij}) - \nu(c_{ij})\right|\right) = 0.76 < \delta$ hay $[R_F^\delta \tilde{\cup} D] \overset{\delta}{\neq} [C \tilde{\cup} D]$, do đó tiếp tục vòng lặp ta có:

- Tính:

$SIG_{R_F^\delta}(a) = \tilde{\tilde{d}}\left([R_F^\delta], [R_F^\delta \tilde{\cup} D]\right) - \tilde{\tilde{d}}\left([R_F^\delta \tilde{\cup} \{a\}], [R_F^\delta \cup \tilde{\{a\}} \cup D]\right) = 0.18 - 0.15 = 0.03$

$SIG_{R_F^\delta}(b) = \tilde{\tilde{d}}\left([R_F^\delta], [R_F^\delta \tilde{\cup} D]\right) - \tilde{\tilde{d}}\left([R_F^\delta \tilde{\cup} \{b\}], [R_F^\delta \cup \tilde{\{b\}} \cup D]\right) = 0.18 - 0.18 = 0.00$

$SIG_{R_F^\delta}(d) = \tilde{\tilde{d}}\left([R_F^\delta], [R_F^\delta \tilde{\cup} D]\right) - \tilde{\tilde{d}}\left(P^{\tilde{\approx}}_{R_F^\delta \cup \{d\}}, [R_F^\delta \tilde{\cup} d \cup D]\right) = 0.18 - 0.18 = 0.00$

$SIG_{R_F^\delta}(f) = \tilde{\tilde{d}}\left([R_F^\delta], [R_F^\delta \tilde{\cup} D]\right) - \tilde{\tilde{d}}\left([R_F^\delta \tilde{\cup} \{f\}], [R_F^\delta \cup \tilde{\{f\}} \cup D]\right) = 0.18 - 0.18 = 0.00$

- Chọn a vì $SIG_{R_F^\delta}(a)$ lớn nhất, do đó

$R_F^\delta = R_F^\delta \cup \{a\} = \{e,c,a\}$.

Vì $\min\left(1 - \sup_{i,j=1}^{6} \left|\mu(b_{ij}) - \mu(c_{ij})\right|, 1 - \sup_{i,j=1}^{6} \left|\nu(b_{ij}) - \nu(c_{ij})\right|\right) = 0.8 = \delta$ hay $[R_F^\delta \tilde{\cup} D] \overset{\delta}{=} [C \tilde{\cup} D]$.

Theo giả thiết ban đầu, tập rút gọn ứng viên $R_F^{0.8}$ có độ chính xác cao nhất trong các ngưỡng δ do đó $R_W^\delta = \{e,c,a\}$. Kết thúc giao đoạn filter - W_δ. Thuật toán chuyển đến giai đoạn W_A của thuật toán IFD.

Giai đoạn W_A:

Tập rút gọn ứng viên R_F^δ được chia thành các tập con $\{e,c\}$ tương ứng với khoảng $[1:2]$ và tập con $\{e,c,a\}$ tương ứng với khoảng $[1:3]$ của tập thuộc tính R_W^δ. Chọn tập con tập thuộc tính có độ chính xác phân lớp cao nhất trên mô hình phân lớp $Model$. Giả sử $\{e,c\}$ là tập thuộc tính con có độ chính xác phân lớp cao nhất, khi đó $R_W^A = \{e,c\}$.

2.3.2. Thực nghiệm và đánh giá thuật toán

Trong các kết quả nghiên cứu của A.Tan và các cộng sự đã chỉ ra phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm hiệu quả hơn tiếp cận tập thô mờ truyền thống về độ chính xác phân lớp. Do đó, chương này sử dụng hai thuật toán của A. Tan [36, 15] để so sánh và đánh giá thuật toán đề xuất IFD. Trong đó thuật toán [36] sử dụng độ đo miền dương mờ trực cảm (Intuitionistic Fuzzy POS - IFPOS[36])và thuật toán [15] sử dụng độ đo Entropy mờ trực cảm(Intuitionistic Fuzzy Information Entropy - IFIE[15]).

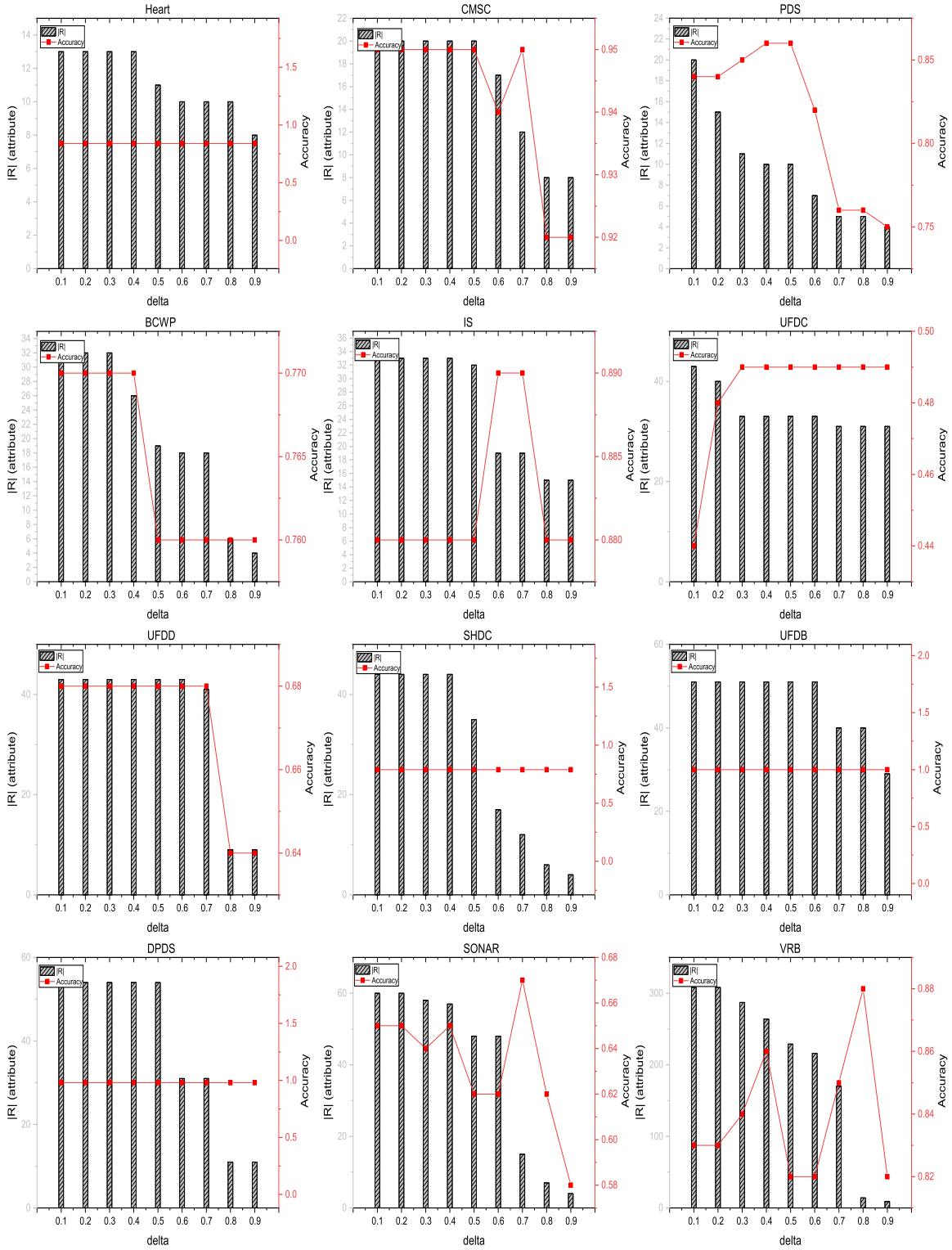
Bảng 2.2: *Bảng mô tả các tập dữ liệu thực nghiệm*

STT	Dataset	Mô tả.	|U|	|C|	|D|
1	Heart	Statlog (Heart)	270	13	2
2	CMSC	Climate Model Simulation Crashes Data Set	540	18	2
3	PDS	Parkinsons Data Set	196	22	2
4	BCWP	Breast Cancer Wisconsin (Prognostic)	198	33	2
5	IS	Ionosphere	351	34	2
6	UFDC	Ultrasonic flowmeter diagnostics (C)	181	43	4
7	UFDD	Ultrasonic flowmeter diagnostics (D)	181	43	4
8	SHDC	SPECTF Heart Data Set	267	44	2
9	UFDB	Ultrasonic flowmeter diagnostics (B)	92	51	3
10	DPDS	Divorce Predictors data set	170	54	2
11	Sona	Connectionist Bench	208	60	2
12	VRB	Voice Rehabilitation(Binary)	126	310	2

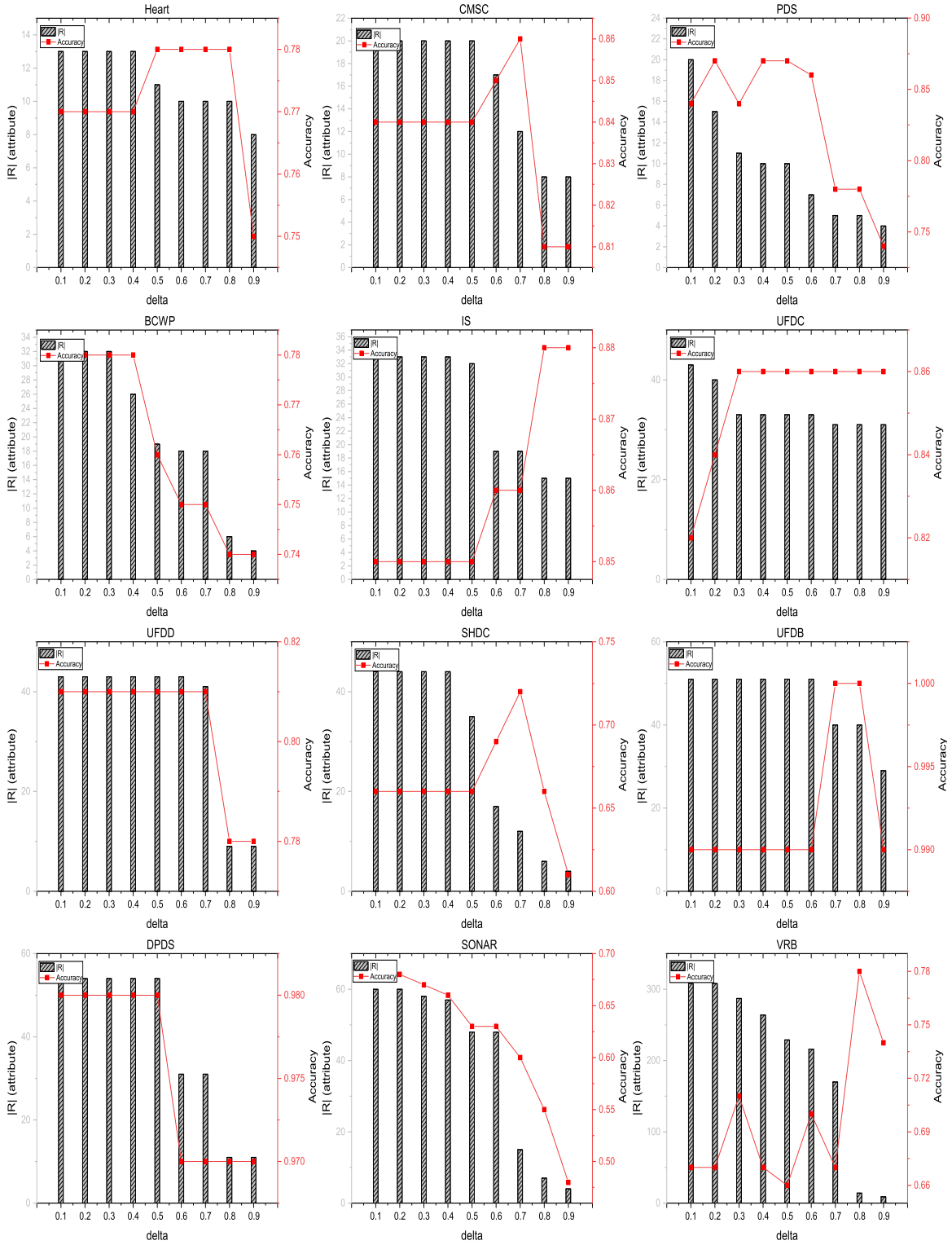
2.3.2.1. Môi trường thực nghiệm

Các thuật toán được cài đặt bằng ngôn ngữ lập trình Pyhton và chạy trên nền hệ điều hành Window 10 với cấu hình phần cứng là bộ xử lý Core I5, ram 8GB. Cùng với 12 tập dữ liệu thử nghiêm được tải về từ UCI được mô tả chi tiết trong Bảng 2.2. Trong đó $|U|$ là số lượng mẫu, $|C|$ là số thuộc tính điều kiện và $|D|$ là số phân lớp của thuộc tính quyết định trong mỗi tập dữ liệu (dataset).

Các tập dữ liệu thử nghiệm đều là các dataset có thuộc tính điều kiện miền giá trị



Hình 2.1: *Tác động của δ tới kích thước và độ chính xác phân lớp trên mô hình phân lớp SVM*



Hình 2.2: *Tác động của δ tới kích thước và độ chính xác phân lớp trên mô hình phân lớp KNN*

số liên tục. Do đó, trước khi thực hiện thuật toán rút gọn thuộc tính, miền giá trị của các thuộc tính sẽ được chuẩn hóa về đoạn [0,1]. Độ tương tự (hàm thuộc) của $x, y \in U$ theo thuộc tính a được xác định bởi công thức $\mu = R_a(x,y) = 1 - |a(x) - a(y)|$, độ khác biệt (hàm không thuộc) được tính theo công thức $\nu = \frac{1-\mu}{1+\lambda\mu}$ với $\lambda > 0$, khi $\lambda = 0$ công thức này suy biến về tập mờ tuyền thống, giá trị của λ càng tăng thì giá trị của ν càng giảm. Tuy nhiên, với sự đa dạng của dữ liệu, việc phân bố dữ liệu trong các tập dữ liệu là rất khác nhau. Do đó để chọn được hệ số λ dùng chung cho toàn bộ các tập dữ liệu là rất khó, đặc biệt là trên các bộ dữ liệu có độ chính xác phân lớp ban đầu thấp. Do đó, chương này đề xuất công thức tính giá trị λ như sau:

$$\lambda_a = \begin{cases} 1 \Leftrightarrow \sigma_a = 0 \\ \frac{\beta_a}{\sigma_a} \Leftrightarrow \sigma_a > 0 \end{cases} \tag{2.4}$$

Trong đó $\sigma_a = \sqrt{\frac{1}{n-1}\sum_{1}^{n}(a(y_i) - \bar{a})^2}$ là độ lệch chuẩn của miền giá trị thuộc tính a và $\beta_a = \frac{\left|P^F_{\{a\}\cup\{d\}}\right|}{\left|P^F_{\{d\}}\right|}$ là độ nhất quán của thuộc tính a trong bảng quyết định.

Để đánh giá độ chinh xác phân lớp của tập rút gọn. Chương này sử dụng hai mô hình phân lớp dữ liệu số là SVM và k-NN(k=|D|). Độ đo đánh giá và phương pháp đánh giá độ chính xác trên các mô hình là độ đo *Accuracy* và phương pháp đánh giá chéo *10-folde* được sử dụng chung cho toàn bộ các tập rút gọn thu được từ các thuật toán.

2.3.2.2. Kịch bản thực nghiệm

Nhằm khẳng định phương pháp rút gọn thuộc tính đề xuất là hiệu quả hơn về độ chính xác phân lớp so với một số phương pháp rút gọn thuộc tính khác của A.Tan và các cộng sự đề xuất, chương này tiến hành thực nghiệm thuật toán đề xuất IFD theo các kịch bản như sau:

1) Lựa chọn giá trị δ tốt nhất cho thuật toán. Thuật toán đề xuất có hai bước wrapper là wrapper_delta (W_delta) và wrapper thuộc tính (W_A), trong đó bước

W_delta được thực hiện trước bước W_A. Như vậy, với mỗi bộ dữ liệu khác nhau sẽ có một giá trị δ khác nhau sao cho tập rút gọn thu được sẽ được tối ưu cả về độ chính xác phân lớp và kích thước.

2) Đánh giá tập rút gọn của thuật toán đề xuất IFD với các thuật toán IFPOS[36], IFIE[15]. Trong đó các tiêu chí được sử dụng để so sánh và đánh giá bao gồm độ chính xác phân lớp (accuracy), kích thước của tập rút gọn ($|R|$) và thời gian thực hiện của thuật toán (second).

2.3.2.3. Lựa chọn giá trị δ

Bảng 2.3: *Mô tả mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn tại hai giai đoạn wrapper trên mô hình phân lớp SVM*

STT	Tập dữ liệu	|U|	|R|			Độ chính xác phân lớp (%)		
			|C|	W_δ	W_A	|C|	W_δ	W_A
1	Heart	270	13	8	7	0.84	0.84	0.84
2	CMSC	540	20	12	11	0.95	0.95	0.95
3	PDS	195	22	10	9	0.84	0.86	0.86
4	BCWP	198	32	26	25	0.77	0.77	0.77
5	IS	351	34	19	16	0.88	0.89	0.89
6	UFDC	181	43	31	26	0.44	0.49	0.52
7	UFDD	180	43	41	27	0.68	0.68	0.68
8	SHDC	267	44	4	2	0.79	0.79	0.79
9	UFDB	92	51	29	2	1	1	1
10	DPDS	170	54	11	5	0.98	0.98	0.98
11	Sonar	208	60	15	11	0.65	0.67	0.7
12	VRB	126	310	14	11	0.83	0.88	0.88

Trước tiên, Để minh họa sự tác động của giá trị δ tới kích thước của tập rút gọn thu được từ thuật toán đề xuất. Chương này xây dựng biểu đồ 2.1 và biểu đồ 2.2 để minh họa sự tác động của δ tới kích thước và độ chính xác phân lớp của tập rút gọn thu được trên mỗi mô hình phân lớp dữ liệu.

Thông quan sự biến động về kích thước và độ chính xác phân lớp của các tập thuộc tính con thu được từ sự thay đổi của các giá trị δ, ta có thể thấy khi giá trị δ càng tăng, kích thước của tập rút gọn càng giảm. Điều đó cho thấy sự thay đổi tuyến tính về kích

Bảng 2.4: *Mô tả mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn tại hai giai đoạn wrapper trên mô hình phân lớp KNN*

STT	Tập dữ liệu	|U|	|R|			Độ chính xác phân lớp (%)		
			|C|	W_δ	W_A	|C|	W_δ	W_A
1	heart	270	13	10	9	0.77	0.78	0.78
2	CMSC	540	20	12	11	0.84	0.86	0.86
3	PDS	195	22	10	7	0.85	0.87	0.87
4	BCWP	198	32	26	21	0.78	0.78	0.79
5	IS	351	34	15	5	0.85	0.88	0.92
6	UFDC	181	43	31	29	0.82	0.86	0.86
7	UFDD	180	43	41	25	0.81	0.81	0.84
8	SHDC	267	44	12	9	0.66	0.72	0.72
9	UFDB	92	51	29	2	0.99	0.99	1
10	DPDS	170	54	11	7	0.98	0.97	0.97
11	sonar	208	60	48	31	0.68	0.63	0.69
12	VRB	126	310	14	12	0.68	0.78	0.82

thước của tập rút gọn so với giá trị thay đổi của δ. Tuy nhiên độ chính xác phân lớp trên cả hai mô hình phân lớp lại không tuyến tính với sự thay đổi của giá trị δ.

2.3.2.4. Đánh giá tập rút gọn của các thuật toán

Bảng 2.3 và Bảng 2.4 mô tả kích thước của các tập rút gọn ứng viên tại giai đoạn W_δ và tập rút gọn tại giai đoạn W_A tương ứng với hai mô hình phân lớp k-NN và SVM. Sau giai đoạn W_δ ta thu được tập rút gọn ứng viên mức delta, sau giai đoạn W_A ta thu được tập rút gọn thực sự của thuật toán IFD. Ta có thể thấy kích thước của tập rút gọn thực sự nhỏ hơn đáng kể so với kích thước của tập rút gọn ứng viên. Đặc biệt trên mô hình phân lớp KNN, tập rút gọn thực sự không những cải thiện cả về kích thước mà độ chính xác của tập rút gọn thu được cũng hiệu quả hơn so với tập rút gọn ứng viên của thuật toán.

Quan sát Bảng 2.5 ta có thể thấy kích thước trung bình của tập rút gọn trên toàn bộ các tập dữ liệu không chênh lệch quá nhiều so với thuật toán rút gọn thuộc tính theo tiếp cận IFPOS[36] nhưng lại tốt hơn so với thuật toán theo tiếp cận IFIE[15]. Trên hai mô hình phân lớp SVM và KNN ta có thể thấy kích thước tập rút gọn được

Bảng 2.5: *Mô tả kích thước thu được của tập rút gọn thu được từ các thuật toán*

STT	Tập rút gọn	|C|	|R|			
			IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
1	heart	13	7	9	13	10
2	CMSC	20	11	11	20	20
3	PDS	22	9	7	8	10
4	BCWP	32	25	21	12	12
5	IS	34	16	5	11	19
6	UFDC	43	26	29	8	11
7	UFDD	43	27	25	6	8
8	SHDC	44	2	9	10	14
9	UFDB	51	2	2	5	11
10	DPDS	54	5	7	15	24
11	sonar	60	11	31	17	25
12	VRB	310	11	12	18	35

wrapper theo mô hình phân lớp SVM là tốt hơn mô hình phân lớp KNN.

Mặc dù kích thước trung bình của các tập rút gọn thu được từ các thuật toán không chênh lệch nhau đáng kể nhưng quan sát trên các bộ dữ liệu có kích thước lớn như **CMSC** và bộ dữ liệu có số chiều lớn như **VRB** ta có thể thấy kích thước tập rút gọn thu được từ thuật toán đề xuất là tốt hơn hẳn so với hai thuật toán còn lại. Để có thể quan sát trực quan hơn sự khác biệt về tập rút gọn thu được từ các thuật toán trên tưng tập dữ liệu, chúng ta có thể quan sát Hình 2.3 và Hình 2.4 để biết thêm thông tin chi tiết.

Bảng 2.6 và Bảng 2.7 mô tả sự chênh lệch về độ chính xác phân lớp của các tập rút gọn tương ứng với các mô hình phân lớp SVM và KNN. Chúng ta có thể thấy trên mô hình phân lớp SVM, thuật toán đề xuất cho tập rút gọn có độ chính xác phân lớp trung bình trên các bộ dữ liệu là tương đương với thuật toán theo tiếp cận IFPOS[36] và tốt hơn thuật toán theo tiếp cận IFE là 3%.

Thuật toán đề xuất IFD và thuật toán IFPOS[36] đều cho tập rút gọn có độ chính xác cao hơn so với tập dữ liệu gốc (Raw). Mặc dù, độ chính xác trung bình trên các bộ dữ liệu rút gọn của hai thuật toán này là như nhau nhưng khả năng cải thiện nhiễu

Bảng 2.6: *So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình phân lớp SVM*

STT	Tập dữ liệu	|U|	Độ chính xác phân lớp (%)			
			Raw	IFD-SVM	IFPOS[36]	IFIE[15]
1	heart	270	84±0.7	84±0	84±0.6	82±0.7
2	CMSC	540	95±0.2	95±0.9	95±0.9	95±0.2
3	PDS	195	84±0.5	85±0.1	85±0.1	84±0.7
4	BCWP	198	77±0.2	77±0.1	76±0.7	76±0.5
5	IS	351	88±0	89±0.9	87±0.6	87±0.6
6	**UFDC**	181	44±0.1	52±0	49±1	49±0.3
7	UFDD	180	68±0.9	68±1	64±0.8	63±0.8
8	SHDC	267	79±0.6	79±0.5	79±0.8	79±0.9
9	UFDB	92	100.0	100.0	100.0	92±0.4
10	DPDS	170	98±0.5	98±0.5	98±0.7	98±0.3
11	**sonar**	208	65±0.3	70±0.5	70±0.2	64±0.7
12	VRB	126	83±0.7	88±0.7	91±0.2	80±0.5

Bảng 2.7: *So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình phân lớp KNN*

STT	Tập dữ liệu	|U|	Độ chính xác phân lớp (%)			
			Raw	IFD-KNN	IFPOS[36]	IFIE[15]
1	heart	270	77±0.4	78±0.2	77±0.6	76±0.8
2	CMSC	540	84±0.9	86±0.9	84±0.4	84±0.6
3	PDS	195	85±0.5	87±0.8	87±0.3	84±0.3
4	BCWP	198	78±0.7	79±0.8	79±0.1	79±0.1
5	IS	351	85±0.3	92±0.5	88±0.6	88±0.6
6	**UFDC**	181	82±0.7	86±0.8	74±0.5	78±0.3
7	UFDD	180	81±0.8	84±0.2	77±0	82±0.1
8	**SHDC**	267	66±0.3	72±0.4	69±0.8	67±0.2
9	UFDB	92	99.0	100.0	100.0	98±0.8
10	DPDS	170	98±1	97±0.2	98±0.5	96±0.8
11	**sonar**	208	68±0.8	69±0.1	62±0.9	60±0.9
12	VRB	126	68±0.6	82±0.7	81±0.7	65±0.2

trên thuật toán đề xuất IFD là tốt hơn thuật toán IFPOS[36]. Cụ thể với bộ dữ liệu **UFDC** có độ chính xác phân lớp ban đầu là 0.44 (44%), thuật toán IFD cho ra tập rút gọn có độ chính xác 0.52 (52%) và thuật toán IFPOS[36] cho tập rút gọn có độ chính xác 0.49 (49%). Trên bộ dữ liệu Sona cả hai thuật toán đều cải thiện nhiễu tốt như

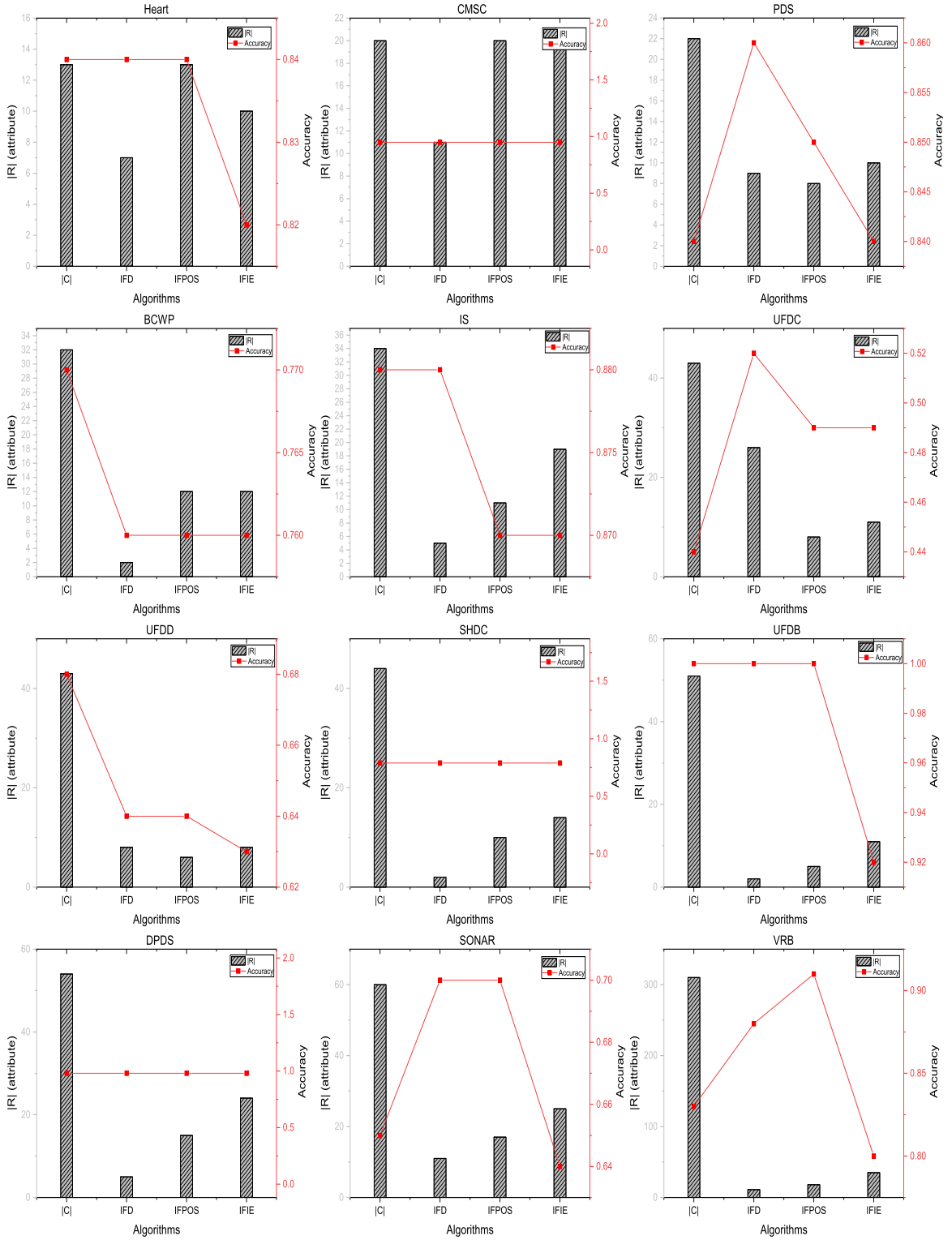
nhau tăng từ 0.65 (65%) lên 0.7 (70%). Đối với mô hình phân lớp KNN, thuật toán đề xuất IFD cho tập rút gọn có độ chính xác phân lớp trung bình trên toàn bộ dữ liệu trội hơn so với hai thuật toán còn lại. Với các bộ dữ liệu có độ chính xác phân lớp ban đầu thấp như **SHDC**, **Sona**, **VRB** đã được cải thiện hiệu quả về độ chính xác phân lớp với thuật toán IFD. Tuy nhiên, Bảng 2.8 cho thấy thời gian thực hiện của thuật toán đề xuất IFD còn thấp hơn so với thuật toán IFPOS[36] và thấp hơn đáng kể so với thuật toán IFIE[15].

Bảng 2.8: *Mô tả thời gian thực hiện của các thuật toán*

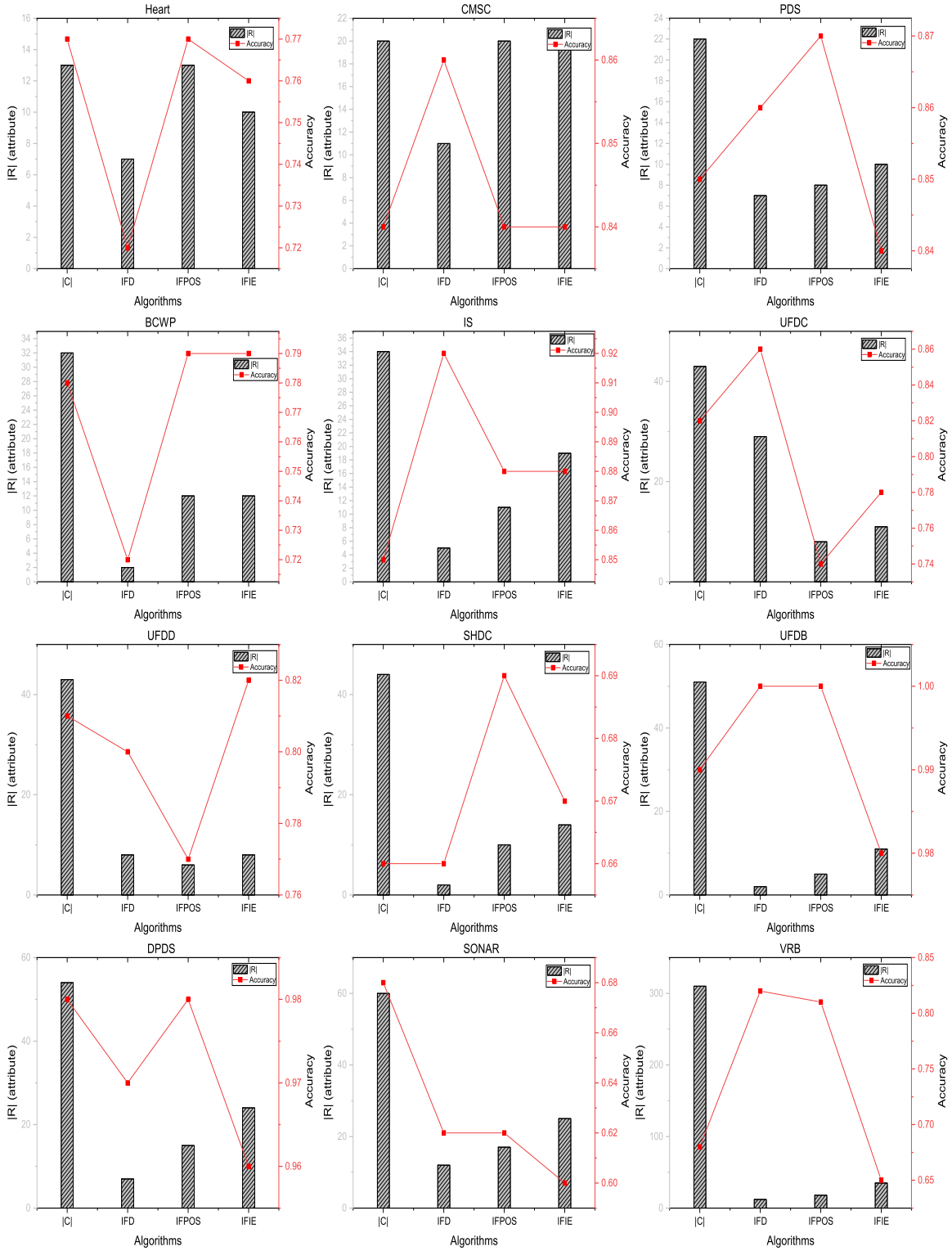
STT	Tập dữ liệu	|U|	|C|	Thời gian thực hiện (s)			
				IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
1	heart	270	13	7.89	7.91	1.58	2.06
2	CMSC	540	18	53.33	54.25	36.56	12.28
3	PDS	195	22	7.67	7.63	6.18	2.46
4	BCWP	198	33	13.09	12.3	8.26	4.28
5	IS	351	34	44.94	43.94	24.34	12.16
6	UFDC	181	43	14.83	13.43	13.46	4.98
7	UFDD	180	43	13.08	13.64	10.86	4.22
8	SHDC	267	44	29.75	32.52	18.32	9.6
9	UFDB	92	51	4.5	5.02	2.78	2.44
10	DPDS	170	54	13.83	16.14	12.84	8.68
11	sonar	208	60	27.12	29.1	23.26	13.3
12	VRB	126	310	52.86	49.83	57.34	57.18

Nhìn chung, tiêu chí độ chính xác phân lớp và kích thước của tập rút gọn của hai thuật toán IFD và IFPOS[36] là tốt nhất. Trong khi đó, hầu hết các thuật toán rút gọn thuộc tính theo tiếp cận độ đo dựa trên tập thô và tập thô mở rộng lâu nay vẫn được chứng minh là tiếp cận bảo toàn thông tin tốt nhất so với các tiếp cận độ đo khác. Điều đó càng khẳng định phương pháp rút gọn thuộc tính theo tiếp cận độ đo khoảng cách mờ trực cảm IFD đề xuất là hiệu quả và đáng được quan tâm. Sau đây là các phân tích về nguyên nhân ảnh hưởng tới thời gian thực hiện của thuật toán IFD, kích thước và độ chính xác phân lớp của tập rút gọn thu được bởi thuật toán IFD

- Kích thước của tập rút gọn: Các kết quả so sánh trong Bảng 2.5 cũng như sự



Hình 2.3: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán trên mô hình phân lớp SVM*



Hình 2.4: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán trên mô hình phân lớp KNN*

chênh lệch của hai giai đoạn wrapper đã được phân tích bên trên cho thấy giai đoạn W_A có vai trò quan trọng trong việc giảm thuộc tính trong khi vẫn giữ được độ chính xác phân lớp tốt nhất của giai đoạn W_δ. Hơn nữa quan sát các Hình 2.1 và 2.2 chúng ta có thể thấy rõ về mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn. Hầu hết các tập dữ liệu được phân tích cho thấy tập rút gọn có kích thước lớn hơn chưa chắc có độ chính xác phân lớp cao hơn so với tập rút gọn có kích thước nhỏ hơn. Đây chính là nguyên nhân chương này đưa giai đoạn W_A vào sau giai đoạn W_δ trong phương pháp rút gọn thuộc tính đề xuất.

- Độ chính xác phân lớp của tập rút gọn: Bên cạnh việc sử dụng tập nền IFS để xây dựng độ đo đánh giá độ quan trọng của thuộc tính. Giai đoạn W_δ của thuật toán rút gọn thuộc tính có ảnh hưởng quan trọng đến độ chính xác phân lớp của tập rút gọn thực sự thu được từ thuật toán. Với mỗi giá trị δ khác nhau, tập rút gọn ứng viên thu được có thể có kích thước và độ chính xác phân lớp khác nhau. Thông qua giai đoạn W_δ, thuật toán đề xuất đã lọc bỏ đi phần lớn các thuộc tính không cần thiết, giảm thời gian tính toán cho giai đoạn W_A của thuật toán đề xuất.

- Khả năng cải thiện nhiễu: Cải thiện nhiễu tốt cũng chính là tăng độ chính xác phân lớp cho tập rút gọn. Bên cạnh yếu tố về tập nền IFS có khả năng cải thiện nhiễu như đã được phân tích trong các công trình nghiên cứu của A.Tan và các cộng sự [36, 15] thi cách thức xây dựng công thức tính độ thuộc và độ không thuộc theo tiếp cận độ nhất quán của thuộc tính trong phần thực nghiệm của Chương cũng ảnh hưởng quan trọng tới việc cải thiện nhiễu, tăng độ chính xác phân lớp cho tập dữ liệu.

- Thời gian tính toán: Thời gian thực hiện của thuật toán rút gọn thuộc tính đề xuất IFD còn hạn chế về mặt thời gian tính toán so với các phương pháp rút gọn thuộc tính khác. Nguyên nhân chính là sự ảnh hưởng của hai giai đoạn wrapper của thuật toán đề xuất. giai đoạn W_δ bị phụ thuộc vào độ phức tạp của độ đo đề xuất và số lượng giá trị δ cần xét. Đây là mối quan hệ tuyến tính. Giai đoạn W_A có thời gian tính toán phụ thuộc vào mô hình phân lớp và kích thước của tập rút gọn ứng viên. Nếu tập rút gọn ứng viên có kích thước nhỏ thì giai đoạn W_A sẽ thực hiện nhanh và ngược lại.

Bảng 2.9: *Mô tả tập rút gọn thu được từ các thuật toán*

STT	Tập dữ liệu	Tập rút gọn thu được từ các thuật toán			
		IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
1	heart	[12, 6, 1, 2, 8, 5, 11]	[12, 6, 1, 2, 8, 5, 11, 10, 9]	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]	[6, 1, 8, 12, 5, 10, 2, 11, 0, 3]
2	CMSC	[0, 2, 1, 19, 3, 18, 17, 4, 16, 15, 14]	[0, 2, 1, 19, 3, 18, 17, 4, 16, 15, 14]	[2, 3, 15, 0, 5, 1, 6, 4, 7, 9, 13, 8, 10, 11, 12, 14, 16, 17, 18, 19]	[0, 1, 4, 6, 5, 7, 10, 8, 9, 3, 11, 17, 12, 2, 13, 14, 15, 16, 18, 19]
3	PDS	[0, 2, 16, 1, 3, 21, 10, 20, 19]	[0, 2, 16, 1, 3, 21, 10]	[18, 0, 10, 16, 17, 2, 19, 20]	[16, 2, 17, 1, 0, 10, 3, 19, 6, 20]
4	BCWP	[0, 1, 31, 6, 30, 29, 11, 28, 27, 26, 25, 12, 24, 23, 22, 19, 21, 20, 18, 17, 16, 15, 14, 13, 10]	[0, 1, 31, 6, 30, 29, 11, 28, 27, 26, 25, 12, 24, 23, 22, 19, 21, 20, 18, 17, 16]	[0, 4, 2, 5, 6, 19, 11, 9, 12, 17, 31, 20]	[0, 31, 6, 19, 1, 11, 22, 5, 20, 8, 12, 18]

Tiếp theo trang sau

Bảng 2.9 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn thu được từ các thuật toán			
		IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
5	IS	[14, 0, 2, 4, 7, 8, 33, 11, 32, 31, 30, 24, 29, 28, 27, 26]	[14, 0, 2, 4, 7]	[0, 4, 2, 5, 27, 30, 7, 3, 9, 16, 17]	[14, 0, 27, 28, 7, 31, 18, 23, 26, 4, 2, 17, 3, 5, 9, 10, 24, 12, 19]
6	UFDC	[3, 23, 27, 8, 18, 42, 41, 40, 39, 38, 37, 36, 35, 34, 0, 33, 32, 31, 30, 2, 29, 28, 26, 25, 24, 22]	[3, 23, 27, 8, 18, 42, 41, 40, 39, 38, 37, 36, 35, 34, 0, 33, 32, 31, 30, 2, 29, 28, 26, 25, 24, 22, 21, 20, 19]	[7, 9, 25, 5, 27, 0, 39, 11]	[3, 23, 8, 25, 0, 2, 27, 15, 29, 33, 31]
7	UFDD	[33, 5, 21, 27, 42, 41, 40, 39, 38, 37, 36, 35, 34, 32, 31, 30, 29, 28, 26, 25, 24, 23, 22, 20, 19, 18, 17]	[33, 5, 21, 27, 42, 41, 40, 39, 38, 37, 36, 35, 34, 32, 31, 30, 29, 28, 26, 25, 24, 23, 22, 20, 19]	[25, 27, 17, 3, 11, 42]	[5, 27, 21, 42, 39, 31, 0, 41]

Tiếp theo trang sau

Bảng 2.9 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn thu được từ các thuật toán			
		IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
8	SHDC	[25, 1]	[25, 1, 2, 21, 43, 42, 41, 29, 40]	[40, 29, 1, 2, 13, 25, 3, 4, 18, 9]	[43, 3, 18, 1, 12, 14, 21, 9, 29, 36, 23, 2, 24, 41]
9	UFDB	[41, 14]	[41, 14]	[41, 14, 13, 16, 12]	[35, 6, 12, 39, 31, 0, 19, 22, 23, 3, 43]
10	DPDS	[39, 34, 0, 53, 15]	[39, 34, 0, 53, 15, 52, 51]	[10, 32, 30, 0, 48, 15, 6, 39, 35, 3, 19, 43, 42, 27, 46]	[44, 45, 3, 51, 34, 5, 30, 46, 41, 6, 48, 2, 36, 52, 43, 0, 38, 21, 42, 1, 31, 47, 53, 27]
11	sonar	[19, 16, 22, 25, 34, 28, 44, 59, 58, 57, 35]	[19, 16, 22, 25, 34, 28, 44, 59, 58, 57, 35, 56, 55, 54, 53, 31, 52, 51, 50, ...]	[0, 11, 15, 36, 26, 19, 21, 9, 53, 23, 24, 28, 6, 30, 32, 35, 44]	[19, 25, 16, 22, 34, 27, 29, 31, 36, 53, 44, 0, 5, 9, 11, 17, 20, 7, 24, 18, 26, 28, ...]

Tiếp theo trang sau

Bảng 2.9 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn thu được từ các thuật toán			
		IFD-SVM	IFD-KNN	IFPOS[36]	IFIE[15]
12	VRB	[59, 70, 1, 69, 34, 41, 62, 79, 57, 309, 83]	[59, 70, 1, 69, 34, 41, 62, 79, 57, 309, 83, 308]	[54, 84, 79, 83, 91, 41, 3, 16, 46, 52, 70, 34, 55, 59, 62, 64, 90, 69]	[58, 70, 59, 60, 62, 92, 55, 69, 127, 57, 107, 121, 138, 25, ...]

2.4. Kết luận Chương 2

Chương 2, luận án trình bày về một phương pháp rút gọn thuộc tính theo tiếp cận IFRS. Các đóng góp chính của Chương này gồm có:

- Đề xuất độ đo khoảng cách mờ trực cảm là cơ sở để xây dựng độ đo đánh giá độ quan trọng của thuộc tính

- Đề xuất thuật toán tìm tập rút gọn trong bảng quyết định số với định nghĩa mới về tập rút gọn theo tiếp cận δ - equal.

Bên cạnh đó, phương pháp xây dựng hàm thành viên và hàm không thành viên cho không gian xấp xỉ mờ trực cảm theo tiếp cận độ nhất quán của thuộc tính do tác giả đề xuất cũng là nhân tố quan trọng ảnh hưởng tới khả năng chọn lọc thuộc tính cho tập rút gọn.

Các kết quả thực nghiệm cho thấy thuật toán đề xuất cho các tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp trên hầu hết các tập dữ liệu so với các thuật toán theo tiếp cận IFRS khác. Tuy nhiên thời gian thực hiện của thuật toán đề xuất còn hạn chế do phải đánh đổi về kích thước và độ chính xác phân lớp cho tập rút gọn, đặc biệt là mục tiêu nâng cao chất lượng phân lớp cho các bộ dữ liệu có độ chính xác phân lớp ban đầu thấp.

CHƯƠNG 3. PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRONG BẢNG QUYẾT ĐỊNH THEO TIẾP CẬN TÔPÔ MỜ TRỰC CẢM

3.1. Mở đầu

Tôpô là một nhánh toán học mà các khái niệm của nó xuất hiện phổ biến trong các lĩnh vực toán học khác và trong các ứng dụng cuộc sống. Theo góc nhìn của lý thuyết tập thô, cấu trúc tôpô là nền tảng toán học quan trọng trong quá trình trích rút, phân tích và xử lý thông tin [38]. Do đó, việc kết hợp lý thuyết tập thô và tôpô sẽ tăng khả năng xử lý trên các tập dữ liệu phức tạp, trong đó các bộ dữ liệu có kích thước lớn, số chiều cao, chứa nhiễu và không đầy đủ xuất hiện ngày càng nhiều trong cuộc sống. Do đó hướng rút gọn thuộc tính theo tiếp cận tôpô trên nền tập thô ngày càng nhận được nhiều quan tâm từ các nhà nghiên cứu [41, 42, 43, 39].

Mặc dù phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm đề xuất đã cải thiện độ độ chính xác phân lớp của tập rút gọn trên các bộ dữ liệu nhiễu nhưng kích thước của tập rút gọn vẫn còn hạn chế. Bên cạnh đó, dựa trên các ưu điểm của cấu trúc tôpô và phần tử đơn vị của tôpô đại số, chương này đề xuất phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm với các bước chính như sau:

- Đề xuất công thức quan hệ ưu tiên mờ trực cảm để xây dựng không gian xấp xỉ mờ trực cảm.

- Đề xuất cấu trúc tôpô mờ trực cảm dựa trên các cơ sở và cơ sở con được định nghĩa theo không gian xấp xỉ mờ trực cảm.

- Đề xuất độ đo tương đồng giữa các tôpô dựa trên khoảng cách tương đồng giữa các cơ sở con, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm tôpô đơn vị, làm cơ sở để định nghĩa tập rút gọn và xây dựng

điều kiện dừng của thuật toán.

Trên cơ sở đó, luận án đề xuất hai phương pháp tìm tập rút gọn như sau:

- Đề xuất phương pháp filter thuộc tính cho tập rút gọn hiệu quả về thời gian và kích thước.

- Đề xuất phương pháp lai ghép filter - wrapper kết hợp cấu trúc dữ liệu ngắn xếp (Stack) cho tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp.

Các kết quả nghiên cứu trong Chương này được công bố trên các công trình nghiên cứu [CT2] và [CT6] đang chờ phản biện vòng 2.

3.2. Đề xuất cấu trúc tôpô mờ trực cảm

Định nghĩa 3.1 (Quan hệ mờ trực cảm)**.** [57] Cho bảng quyết định $DT = (U, C, D, f)$ và quan hệ mờ trực cảm IFR xác định trên U. Khi đó IFR được gọi là quan hệ tương đương mờ trực cảm nếu các điều kiện sau đây thỏa mãn:

(1) Tính phản xạ: $IFR(x,x) = 1_{IF}$ với mọi $x \in U$.

(2) Tính đối xứng: $IFR(x,y) = IFR(y,x)$ với mọi $x, y \in U$.

(3) Tính bắc cầu: $R(x,y) \wedge R(y,z) \leq R(x,z)$ với mọi $x, y, z \in U$.

Khi đó, IFR được gọi là quan hệ ưu tiên mờ trực cảm nếu các tính chất (1) và (3) thỏa mãn.

Định nghĩa 3.2 (Công thức quan hệ mờ trực cảm)**.** Cho bảng quyết định $DT = (U, C, D, f)$, với mọi $(x,y) \in U$ và $\delta \in [0.5, 1]$, Khi đó $IFR_a^{\geq}(x,y) = \langle y, \mu_y, \nu_y \rangle$ với $a \in C$ được xác định bởi:

$$\mu_y = \begin{cases} 1 - |a(x) - a(y)| \text{ nếu } p_a(x,y) \geq \delta \\ 0, \text{còn lại} \end{cases} \qquad (3.1)$$

$$\nu_y = 1 - \mu_y$$

Trong đó $p_a(x,y) = \frac{\mathrm{a}(x) - a(y) + 1}{2}$. Khi đó, giá trị p_a luôn thuộc đoạn $[0.5, 1]$. Khi giá trị $\delta = 0.5$, quan hệ ưu tiên này có tính chất phản xạ và bắc cầu, khi $\delta > 0.5$ quan hệ

ưu tiên này chỉ có tính bắc cầu.

Định nghĩa 3.3 (Ma trận quan hệ ưu tiên mờ trực cảm)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và quan hệ ưu tiên mờ trực cảm $IFR_a^{\geq}$ tương ứng với thuộc tính $a \in C$ xác định trên U. Khi đó $IFR_a^{\geq}$ có thể được biểu diễn bởi ma trận quan hệ $M_a^{\geq} = [i, j]_{n \times n}$.

Ví dụ 3.1. Xét bảng quyết định cho trong Bảng 1.3, với $\delta = 0.5$, ta có:

$$M_a^{\geq} = \begin{bmatrix} (1,0) & (1,0) & (0.8,0.2) & (0.2,0.8) & (0.2,0.8) & (0.2,0.8) \\ (1,0) & (1,0) & (0.8,0.2) & (0.2,0.8) & (0.2,0.8) & (0.2,0.8) \\ (0,1) & (0,1) & (1,0) & (0.4,0.6) & (0.4,0.6) & (0.4,0.6) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \end{bmatrix}$$

Định nghĩa 3.4 (Hợp hai ma trận)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai ma trận quan hệ ưu tiên mờ trực cảm $M_a^{\geq}, M_b^{\geq}$ tương ứng của $a, b \in C$ xác định trên U. Khi đó hợp của hai ma trận được định nghĩa bởi

$$M_a^{\geq} \cup M_b^{\geq}[i, j] = max\left(M_a^{\geq}[i, j], M_b^{\geq}[i, j]\right) \tag{3.2}$$

Định nghĩa 3.5 (Giao hai ma trận)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai ma trận quan hệ ưu tiên mờ trực cảm $M_a^{\geq}, M_b^{\geq}$ tương ứng của $a, b \in C$ xác định trên U. Khi đó giao của hai ma trận được định nghĩa bởi

$$M_a^{\geq} \cap M_b^{\geq}[i, j] = min\left(M_a^{\geq}[i, j], M_b^{\geq}[i, j]\right) \tag{3.3}$$

Định nghĩa 3.6 (Cở sở con IF-subbase)**.** Cho bảng quyết định $DT = (U, C, D, f)$. Khi đó IF-subbase của $a \in C$ được định nghĩa bởi:

$$S_a = \left\{S_a^L, S_a^R\right\} \tag{3.4}$$

Trong đó S_a^L và S_a^R lần lượt là IF-subbase trái tương ứng với ma trận quan hệ $M_a^{\geq}$ và IF-subbase phải tương ứng với ma trận quan hệ $\left(M_a^{\geq}\right)^T$ trên thuộc tính $a \in C$, với $\left(M_a^{\geq}\right)^T$ là ma trận chuyển vị của ma trận $M_a^{\geq}$.

Ví dụ 3.2. Tiếp theo Ví dụ 3.1, thực hiện phép lấy đối xứng ta có:

$$\left(M_a^{\geq}\right)^T = \begin{bmatrix} (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (0.8,0.2) & (0.8,0.2) & (1,0) & (0,1) & (0,1) & (0,1) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \end{bmatrix}$$

Định nghĩa 3.7 (Giao hai IF-subbase)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai IF-subbases S_p, S_q tương ứng với $p, q \in C$. Khi đó, phép toán giao của hai IF-subbase được định nghĩa bởi:

$$S_p \cap S_q = \left\{S_p^L \cap S_q^L, S_p^R \cap S_q^R\right\} \tag{3.5}$$

Định nghĩa 3.8 (Hợp hai IF-subbase)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai IF-subbases S_p, S_q tương ứng với $p, q \in C$. Khi đó, phép toán hợp của hai IF-subbase được định nghĩa bởi:

$$S_p \cup S_q = \left\{S_p^L \cup S_q^L, S_p^R \cup S_q^R\right\} \tag{3.6}$$

Định nghĩa 3.9 (Cơ sở IF-base)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và IF-subbase $S_a = \left\{S_a^L, S_a^R\right\}$ tương ứng với $a \in C$, trong đó S_a^L được gọi là IF-subbase trái và S_a^R được gọi là IF-subbase phải. Khi đó IF-base B_a được định nghĩa bởi:

$$B_a = {S_a}^L \cap {S_a}^R \tag{3.7}$$

Định nghĩa 3.10 (Tôpô mờ trực cảm IFT)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và IF-base B_a tương ứng với $a \in C$. Khi đó IFT $\mathscr{T}_a$ được định nghĩa bởi:

$$\mathscr{T}_a = \{c : c = \cup\{b \in B\}, B \subseteq B_a\} \tag{3.8}$$

Mệnh đề 3.1 (IFT từ IF-base)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và B_a là một IF-base được xác định bởi công thức 3.7. Khi đó, B_a là một cơ sở của $\mathscr{T}_a$.*

Chứng minh. Ta có hai điều phải chứng minh

(1) Theo định nghĩa 3.9 ta có $B_a = S_a^L \cap S_a^R$ là một tập các quan hệ mờ trực cảm chỉ có tính chất phản xạ và đối xứng do IF-subbase trái S_a^L và IF-subbase phải S_a^R là các thành phần đối xứng nhau.

(2) Theo định nghĩa 3.10 về cấu trúc tôpô IFT trên cơ sở IF-base và định nghĩa 1.13 về cơ sở của một tôpô ta thấy rõ ràng B_a là một IF-base.

Từ (1) và (2) ta có điều phải chứng minh (đpcm) □

Ví dụ 3.3. Tiếp theo ví dụ 3.2, thực hiện phép toán giao của hai ma trận đối xứng ta có: $B_a = \begin{bmatrix} (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (0,1) & (0,1) & (1,0) & (0,1) & (0,1) & (0,1) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \end{bmatrix}$

Mệnh đề 3.2 (So sánh hai IF-tôpô)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và hai tôpô $\mathscr{T}_p$, $\mathscr{T}_q$ tương ứng của $p, q \in C$. Khi đó, $\mathscr{T}_p \prec \mathscr{T}_q$ nếu $B_p \prec B_q$.*

Chứng minh. Theo định nghĩa cấu trúc IF-tôpô trên IF-base ta có đpcm. □

Định nghĩa 3.11 (IF-subbase của tập thuộc tính)**.** Cho bảng quyết định $DT = (U, C, D, f)$, với mọi $p, q \in C$. Khi đó IF-subbase của $\{p\} \cup \{q\}$ được định nghĩa bởi:

$$S_{\{p\}\cup\{q\}} = S_p \cap S_q \tag{3.9}$$

Mệnh đề 3.3 (So sánh hai IF-subbase)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và hai IF-subbases S_P, S_Q tương ứng với $P, Q \subseteq C$. Khi đó $S_Q \prec S_P$ nếu $P \subseteq Q$.*

Chứng minh. Theo định nghĩa 3.8 và phép toán giao trên tập nền IFS ta có đpcm. □

Định nghĩa 3.12 (IF-base mịn nhất)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và IF-base B_a tương với $a \in C$. Khi đó B_a được gọi là IF-base mịn nhất (smoothest) nếu:

$$B_a[i,j] = \begin{cases} 1_{IF} \text{ nếu } i = j \\ 0_{IF}, \text{còn lại} \end{cases}$$

Trong đó $1_{IF} = (1,0)$ và $0_{IF} = (0,1)$. Kí hiệu IF-base mịn nhất là B_I là cơ sở của tôpô đơn vị mờ trực cảm.

3.3. Đề xuất độ đo tương đồng của hai tôpô mờ trực cảm

Dựa theo phương pháp xây dựng độ đo khoảng cách giữa hai phân hoạch mờ trực cảm, luận án mở rộng định nghĩa cho độ đo khoảng cách giữa hai cơ sở con mờ trực cảm như sau:

Định nghĩa 3.13 (Khoảng cách giữa hai IF-subbase)**.** Cho bảng quyết định $DT = (U,C,D,f)$ và hai IF-subbases S_p, S_q tương ứng với $p,q \in C$. Khi đó, độ khác biệt giữa S_p và S_q được định nghĩa bởi:

$$\begin{aligned} \zeta(S_p,S_q) = &\frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|S_p^L[i] \cup S_q^L[i]\right| - \left|S_p^L[i] \cap S_q^L[i]\right|\right) \\ &+ \frac{1}{|U|^2}\sum_{i=1}^{|U|}\left(\left|S_p^R[i] \cup S_q^R[i]\right| - \left|S_p^R[i] \cap S_q^R[i]\right|\right) \end{aligned} \tag{3.10}$$

Mệnh đề 3.4 (Độ khác biệt giữa hai IF-subbase)**.** *Cho bảng quyết định $DT = (U,C,D,f)$ và hai IF-subbases S_p, S_q tương ứng với $p,q \in C$. Khi đó:*

$$\zeta(S_p,S_q) = \frac{2}{|U|^2}\sum_{i=1}^{|U|}\left(\left|S_p^L[i] \cup S_q^L[i]\right| - \left|S_p^L[i] \cap S_q^L[i]\right|\right) \tag{3.11}$$

Là độ khác biệt giữa S_p và S_q

Chứng minh. Vì S_p^L, S_p^R, và S_q^L, S_q^R là đối xứng lẫn nhau, do đó $\left|S_p^L\right| = \left|S_p^R\right|$ và $\left|S_q^L\right| = \left|S_q^R\right|$. Ta có đpcm. □

Mệnh đề 3.5 (Độ phụ thuộc của thuộc tính theo IF-subbase)**.** *Cho bảng quyết định*

$DT = (U, C, D, f)$ và hai IF-subbases S_C and $S_{C \cup D}$ tương ứng với C và $C \cup D$. Khi đó:

$$\zeta\left(S_C, S_{C \cup D}\right) = \frac{2}{|U|^2} \sum_{i=1}^{|U|} \left(\left|S_D^L[i] - S_D^L[i] \cap S_C^L[i]\right|\right) \tag{3.12}$$

Là độ phụ thuộc của thuộc tính D với thuộc tính C.

Chứng minh. Theo mệnh đề 3.4 ta có:

$$\begin{aligned}
\zeta\left(S_D, S_{D \cup C}\right) &= \frac{2}{|U|^2} \sum_{i=1}^{|U|} \left(\left|S_D^L[i] \cup S_{D \cup C}^L[i]\right| - \left|S_D^L[i] \cap S_{D \cup C}^L[i]\right|\right) \\
&= \frac{2}{|U|^2} \sum_{i=1}^{|U|} \left(\left|S_D^L[i] \cup \left(S_D^L[i] \cap S_C^L[i]\right)\right| - \left|S_D^L[i] \cap \left(S_D^L[i] \cap S_C^L[i]\right)\right|\right) \\
&= \frac{2}{|U|^2} \sum_{i=1}^{|U|} \left(\left|S_D^L[i]\right| - \left|S_D^L[i] \cap S_C^L[i]\right|\right)
\end{aligned}$$

Ta có đpcm. □

Mệnh đề 3.6 (Tính chất phản đơn điệu của độ đo tương đồng)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và hai IF-subbases S_B, S_C tương ứng với B và C. Khi đó, nếu $B \subseteq C$ thì $\zeta\left(S_D, S_{D \cup C}\right) \leq \zeta\left(S_D, S_{D \cup B}\right)$:*

Chứng minh. Theo mệnh đề 3.3, vì $B \subseteq C$, $S_C^L \leq S_B^L$, nghĩa là với mọi $x \in U$, nếu $[x]_C^L \subseteq [x]_B^L$ thì $\left|[x]_C^L\right| \leq \left|[x]_B^L\right|$. Khi đó $\zeta\left(S_D, S_{D \cup C}\right) \leq \zeta\left(S_D, S_{D \cup B}\right)$. □

3.4. Rút gọn thuộc tính trong bảng quyết định theo tiếp cận tôpô mờ trực cảm

3.4.1. Đề xuất thuật toán tìm tập rút gọn trong bảng quyết định theo phương pháp filter, sử dụng cấu trúc tôpô mờ trực cảm

Định nghĩa 3.14 (Độ quan trọng của thuộc tính)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và tập thuộc tính $R \subseteq C$. Khi đó, độ quan trọng của thuộc tính $a \in C - R$ với tập thuộc tính R được định nghĩa bởi:

$$Sig_R(a) = \zeta\left(S_D, S_{D \cup R \cup \{a\}}\right) - \zeta\left(S_D, S_{D \cup R}\right) \tag{3.13}$$

Mệnh đề 3.7 (Tính tồn tại của tập rút gọn)**.** *Cho bảng quyết định* $DT = (U, C, D, f)$ *và hai IF-bases* B_R *và* B_C *tương ứng với* $R \subseteq C$*. Khi đó, nếu* $B_R = B_I$ *thì* $B_C = B_I$*.*

Chứng minh. Vì $R \subseteq C$, $S_C = S_{R \cup \{C-R\}} = S_R \cap S_{\{C-R\}}$, nghĩa là $B_C = B_R \cap B_{\{C-R\}}$. Khi đó $B_R = B_I \rightarrow B_C = B_I \cap B_{\{C-R\}} = B_I$. □

Dựa trên mệnh đề 3.9 ta có thể khẳng định, nếu một bảng quyết định tồn tại một tập con R của tập thuộc tính ban đầu C mà B_R là cơ sở mịn nhất thì chắc chắn B_C cũng là cơ sở mịn nhất. Nghĩa là $B_R = B_C = B_I$. Khi đó, tập rút gọn theo tiếp cận tôpô có thể được định nghĩa như sau:

Định nghĩa 3.15 (Tập rút gọn theo tiếp cận tôpô đơn vị)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và $R \subseteq C$. Khi đó R được gọi là một tập rút gọn của C khi và chỉ khi

(1) $B_R = B_I$

(2) $B_{R-c} \neq B_I$ với mọi $c \in R$

Để đảm bảo tính tồn tại của B_I, quan hệ ưu tiên mờ trực cảm đề xuất phải có tính chất phản xạ, do đó giá trị δ được chọn mặc định là 0.5 cho toàn bộ các ví dụ minh họa và thực nghiệm các thuật toán. Sau đây là phần đề xuất thuật toán F_IFT tìm tập rút gọn theo phương pháp filter.

Tiếp theo sẽ là phần đánh giá độ phức tạp của thuật toán. Trước tiên, kí hiệu $|U|, |C|$ lần lượt là số lượng các đối tượng và số lượng các thuộc tính của bảng quyết định $DT = (U, C, D, f)$.

(1) Độ phức tạp tại các bước 4-6 là $\mathcal{O}\left(|C|\,|U|^2\right)$;

(2) Độ phức tạp tại các bước 8-10 là $\mathcal{O}\left(|C-R|\,|U|^2\right)$, độ phức tạp tại bước 11 là $\mathcal{O}(|C-R|)$, độ phức tạp tại bước 13 là$\mathcal{O}\left(|U|^2\right)$. Do đó, độ phức tạp tại các bước 7-14 là $\mathcal{O}\left(|R|\,|C-R|\,|U|^2\right)$;

Từ (1) và (2) ta có độ phức tạp của thuật toán F_IFT algorithm là $\mathcal{O}\left(|R|\,|C-R|\,|U|^2\right)$.

Ví dụ 3.4. Để minh họa quá trình hoạt động của thuật toán đề xuất, sau đây là phần trình bày ví dụ số cho thuật toán F_IFT.

Thuật toán 3.1 Rút gọn thuộc tính theo phương pháp filter sử dụng tiếp cận tôpô mờ trực cảm (F_IFT)

Input: Bảng quyết định $DT = (U, C, D, f)$ và $\delta = 0.5$
Output: Tập rút gọn R

1: $R \leftarrow \emptyset$;
2: B_R là cơ sở mờ trực cảm thô nhất;
3: B_I là cơ sở mờ trực cảm mịn nhất;
4: **for all** $c \in C \cup D$ **do**
5: **calculate** S_c; {theo công thức 3.1 và 3.4}
6: **end for**
7: **while** $B_R \neq B_I$ **do**
8: **for all** $c \in C - R$ **do**
9: **calculate** $Sig_R(c)$; {theo công thức 3.13}
10: **end for**
11: **select** $c_m \in C - R : Sig_R(c_m) = \underset{c \in C-R}{Max} \{Sig_R(c)\}$;
12: $R \leftarrow R \cup \{c_m\}$;
13: **update** B_R; {theo công thức 3.7}
14: **end while**
15: **return** R;

Cho bảng quyết định $DT = (U, C, D, f)$ được trình bày như trong Bảng 1.3 trong đó $U = \{u_1, u_2, u_3, u_4, u_5, u_6\}$ và $C = \{a, b, c, d, e, f\}$.

Giai đoạn khởi tạo:

$ST \leftarrow \emptyset$; $R_W \leftarrow \emptyset$; $R_F \leftarrow \emptyset$; $R \leftarrow \emptyset$; B_R là cơ sở mờ trực cảm thô nhất; B_I là cơ sở mờ trực cảm mịn nhất;

Giai đoạn tính các IF-subbase ban đầu S_a, S_b , S_c , S_d , S_e , S_f;

Giai đoạn filter:

Vì $B_R \neq B_I$ do đó thực hiện tính toán độ quan trọng của từng thuộc tính theo IF-subbase như sau:

$$Sig_R(a) = \zeta\left(S_D, S_{D \cup R \cup \{a\}}\right) - \zeta\left(S_D, S_{D \cup R}\right) = 0.23$$

$$Sig_R(b) = \zeta\left(S_D, S_{D \cup R \cup \{b\}}\right) - \zeta\left(S_D, S_{D \cup R}\right) = 0.21$$

$$Sig_R(c) = \zeta\left(S_D, S_{D \cup R \cup \{c\}}\right) - \zeta\left(S_D, S_{D \cup R}\right) = 0.22$$

$$Sig_R(d) = \zeta\left(S_D, S_{D \cup R \cup \{d\}}\right) - \zeta\left(S_D, S_{D \cup R}\right) = 0.19$$

$Sig_R(e) = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.21$

$Sig_R(f) = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.21$

Vì $Sig_R(a)$ lớn nhất nên $R = R\cup\{a\} = \{a\}$ và cập nhật lại B_R. Vì $B_R \neq B_I$, khi đó tiếp tục tính độ quan trọng của các thuộc tính còn lại ta có:

$Sig_R(b) = \zeta\left(S_D, S_{D\cup R\cup\{b\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.08$

$Sig_R(c) = \zeta\left(S_D, S_{D\cup R\cup c}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.04$

$Sig_R(d) = \zeta\left(S_D, S_{D\cup R\cup d}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.02$

$Sig_R(e) = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_R(f) = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.05$

Vì: $Sig_R(b)$ lớn nhất nên $R = R\cup\{a\} = \{a,b\}$, và cập nhật lại B_R. Vì $B_R \neq B_I$ khi đó tiếp tục tính độ quan trọng của các thuộc tính còn lại ta có:

$Sig_c = \zeta\left(S_D, S_{D\cup R\cup\{c\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_d = \zeta\left(S_D, S_{D\cup R\cup\{d\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_e = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_f = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

Vì $Sig_R(c) = Sig_R(d) = Sig_R(e) = Sig_R(f)$ do đó ta chọn c. Khi đó $R = R\cup\{c\} = \{a,b,c\}$. Cập nhật lại B_R ta có $B_R = B_I$. Khi đó vòng lặp kết thúc. Ta có $R = \{a,b,c\}$ là tập rút gọn của thuật toán F_IFT algorithm.

3.4.2. Đề xuất thuật toán tìm tập rút gọn trong bảng quyết định theo phương pháp lai ghép filter - wrapper, sử dụng cấu trúc tôpô mờ trực cảm

Về cơ bản, phương pháp chọn lọc thuộc tính được sử dụng trong thuật toán F_IFT vẫn theo tiếp cận độ đo, do đó khả năng phân loại giữa các thuộc tính vẫn còn thấp. Khi đó, trong quá trình đánh giá, có thể xuất hiện nhiều thuộc tính có cùng độ quan trọng như nhau dẫn tới bỏ sót các thuộc tính ứng viên có thể tốt hơn trong thực tế.

Để giải quyết hạn chế của thuật toán F_IFT, sau đây là phần đề xuất thuật toán lai ghép filter - wrapper FW_IFT tìm tập rút gọn với cấu trúc dữ liệu **Stack** được sử

dụng. Trong đó các tập thuộc tính ứng viên tại giai đoạn **filter** sẽ được đẩy vào **Stack** để sinh các tập rút gọn ứng viên cho giai đoạn **wrapper** của thuật toán.

Thuật toán 3.2 Phương pháp rút gọn thuộc tính lai ghép filter - wrapper sử dụng tiếp cận tôpô mờ trực cảm (FW_IFT)

Input: Bảng quyết định $DT = (U, C, D, f)$ và $\delta = 0.5$, mô hình phân lớp $Model$
Output: Tập rút gọn R

1: $ST \leftarrow \emptyset$; $R_W \leftarrow \emptyset$; $R_F \leftarrow \emptyset$; $R \leftarrow \emptyset$;
2: B_{R_F} là cơ sở mờ trực cảm thô nhất;
3: B_I là cơ sở mờ trực cảm mịn nhất;
4: **for all** $c \in C \cup D$ **do**
5: **calculate** S_c; {theo công thức 3.1 và 3.4}
6: **end for**
7: **for all** $c \in C - R_F$ **do**
8: calculate $Sig_{R_F}(c)$; {theo công thức 3.13}
9: **end for**
10: **for all** $c_m \in \{\underset{c \in C - R_F}{Max} \{Sig_{R_F}(c)\}\}$ **do**
11: $ST.PUSH(R_F \cup \{c_m\})$; {Đẩy c_m vào Stack}
12: **end for**
13: **while** $ST \neq \emptyset$ **do**
14: $R_F = ST.POP$; {giai đoạn filter}
15: **update** B_{R_F}
16: **if** $B_{R_F} = B_I$ **then**
17: $R_W = R_W \cup \{R_F\}$; {Đưa tập rút gọn ứng viên vào danh sách}
18: **else**
19: quay lại bước 10;
20: **end if**
21: **end while**
22: **for all** $r \in R_W$ **do**
23: **if** $ACC(Model, r) > ACC(Model, R)$ **then**
24: $R = r$; {Giai đoạn wrapper}
25: **end if**
26: **end for**
27: **return** R;

Trong đó ST là cấu trúc dữ liệu ngăn xếp Stack với các phép toán PUSH để đẩy dữ liệu vào và POP để đẩy dữ liệu ra khỏi Stack. R_F là tập rút gọn ứng viên tại bước filter của thuật toán, R_W là danh sách chứa các tập rút gọn ứng viên cho giai đoạn wrapper của thuật toán.

Tiếp theo sẽ là phần đánh giá độ phức tạp của thuật toán. Trước tiên, kí hiệu $|U|, |C|$ lần lượt là số lượng các đối tượng và số lượng các thuộc tính của bảng quyết định $DT = (U, C, D, f)$.

(1) Độ phức tạp tại các bước 4-6 là $\mathcal{O}\left(|C||U|^2\right)$;

(2) Độ phức tạp tại các bước 7-9 là $\mathcal{O}\left(|C - R_F||U|^2\right)$;

(3) Độ phức tạp tại các bước 10-12 là $\mathcal{O}(|C - R_F|)$;

(4) Độ phức tạp tại các bước 13-21 là $\mathcal{O}\left(|ST||C - R_F||U|^2\right)$;

(5) From (1), (2), (3), và (4), ta có độ phức tạp của thuật toán FW_IFT tại giai đoạn filter là $\mathcal{O}\left(|ST||C - R_F||U|^2\right)$;

(6) Giả sử độ phức tạp của mô hình phân lớp *Model* là $\mathcal{O}(|T|)$. Khi đó, độ phức tạp tại bước 22-26 là $\mathcal{O}(|R_{\mathrm{W}}||T|)$;

Từ (5) và (6), ta có độ phức tạp của thuật toán FW_IFT là: $\mathcal{O}\left(|ST||C - R_F||U|^2\right) + \mathcal{O}(|R_{\mathrm{W}}||T|)$.

Ví dụ 3.5. Để minh họa quá trình hoạt động của thuật toán đề xuất, sau đây sẽ là phần trình bày ví dụ số cho thuật toán FW_IFT.

Cho bảng quyết định $DT = (U, C, D, f)$ được trình bày như trong Bảng 1.3 trong đó $U = \{u_1, u_2, u_3, u_4, u_5, u_6\}$ và $C = \{a, b, c, d, e, f\}$.

Giai đoạn khởi tạo

$ST \leftarrow \emptyset$; $R_W \leftarrow \emptyset$; $R_F \leftarrow \emptyset$; $R \leftarrow \emptyset$; B_{R_F} là cơ sở mờ trực cảm thô nhất; B_I là cơ sở mờ trực cảm mịn nhất;

Giai đoạn tính các IF-subbase ban đầu S_a, S_b , S_c , S_d , S_e , S_f;

Tính độ quan trọng của từng thuộc tính trong $C - R_F$ ta có:

$$Sig_{R_F}(a) = \zeta\left(S_D, S_{D\cup R\cup\{a\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.23$$

$$Sig_{R_F}(b) = \zeta\left(S_D, S_{D\cup R\cup\{b\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.21$$

$$Sig_{R_F}(c) = \zeta\left(S_D, S_{D\cup R\cup\{c\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.22$$

$$Sig_{R_F}(d) = \zeta\left(S_D, S_{D\cup R\cup\{d\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.19$$

$$Sig_{R_F}(e) = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.21$$

$Sig_{R_F}(f) = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.21$

Vì $Sig_{R_F}(a)$ là lớn nhất nên thực hiện $ST.PUSH(R_F \cup \{a\})$ do đó $ST = \{a\}$

Giai đoạn filter

Vì $ST = \{a\} \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a\}$. Khi đó $ST = \emptyset$ và $B_{R_F} \neq B_I$ nên tính độ quan trọng của các thuộc tính còn lại trong $C - R_F$ với R_F ta có:

$Sig_{R_F}(b) = \zeta\left(S_D, S_{D\cup R\cup\{b\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.08$

$Sig_{R_F}(c) = \zeta\left(S_D, S_{D\cup R\cup\{c\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.04$

$Sig_{R_F}(d) = \zeta\left(S_D, S_{D\cup R\cup\{d\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.02$

$Sig_{R_F}(e) = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_{R_F}(f) = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.05$

Vì $Sig_{R_F}(b)$ là lớn nhất nên thực hiện $ST.PUSH(R_F \cup \{a\})$ do đó $ST = \{a, b\}$

Vì $ST = \{a, b\} \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a\}$. Khi đó $ST = \emptyset$ và $B_{R_F} \neq B_I$ nên tính độ quan trọng của các thuộc tính còn lại trong $C - R_F$ với R_F ta có:

$Sig_c = \zeta\left(S_D, S_{D\cup R\cup\{c\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_d = \zeta\left(S_D, S_{D\cup R\cup\{d\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_e = \zeta\left(S_D, S_{D\cup R\cup\{e\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

$Sig_f = \zeta\left(S_D, S_{D\cup R\cup\{f\}}\right) - \zeta\left(S_D, S_{D\cup R}\right) = 0.01$

Vì $Sig_{R_F}(c) = Sig_{R_F}(d) = Sig_{R_F}(e) = Sig_{R_F}(f)$ nên ta thực hiện các lệnh:

$ST.PUSH(R_F \cup \{c\})$; $ST.PUSH(R_F \cup \{d\})$;

$ST.PUSH(R_F \cup \{e\})$; $ST.PUSH(R_F \cup \{f\})$;

Khi đó $ST = \{\{a, b, c\}; \{a, b, d\}; \{a, b, e\}; \{a, b, f\}\}$

Vì $ST \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a, b, f\}$.

Khi đó $ST = \{\{a, b, c\}; \{a, b, d\}; \{a, b, e\}\}$.

Vì $B_{R_F} = B_I$ do đó $R_W = R_W \cup R_F = \{\{a, b, f\}\}$

Vì $ST \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a, b, e\}$.

Khi đó $ST = \{\{a, b, c\}; \{a, b, d\}\}$.

Vì $B_{R_F} = B_I$ do đó $R_W = R_W \cup R_F = \{\{a,b,f\};\{a,b,e\}\}$

Vì $ST \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a,b,d\}$.

Khi đó $ST = \{\{a,b,c\}\}$.

Vì $B_{R_F} = B_I$ do đó $R_W = R_W \cup R_F = \{\{a,b,f\};\{a,b,e\};\{a,b,d\}\}$

Vì $ST \neq \emptyset$ nên thực hiện $R_F = ST.POP = \{a,b,c\}$ do đó $ST = \emptyset$. Vì $B_R = B_I$ do đó:

$R_W = R_W \cup R_F = \{\{a,b,f\};\{a,b,e\};\{a,b,d\};\{a,b,c\}\}$

Vì $ST = \emptyset$ nên kết thúc giai đoạn filter, chuyển sang giai đoạn wrapper.

Giai đoạn wrapper

Thực hiện đánh giá độ chính xác phân lớp của từng tập rút gọn ứng viên $r \in R_W$. Xác định ứng viên nào có độ chính xác phân lớp cao nhất trên mô hình $Model$. Giả sử ứng viên $r = \{a,b,e\}$ có độ chính xác phân lớp cao nhất, khi đó tập rút gọn thức sự của thuật toán FW_IFT thu được là $R = \{a,b,e\}$.

3.4.3. Thực nghiệm và đánh giá các thuật toán

Phần này sẽ trình bày một số kết quả thực nghiệm của hai thuật toán đề suất F_IFT và FW_IFT trên một số bộ dữ liệu của UCI. Mục tiêu của thực nghiệm nhằm củng cố giả thiết phương pháp rút gọn thuộc tính theo tiếp cận tôpô cho tập rút gọn tối ưu hơn tiếp cận độ đo truyền thống [39]. Trong đó thuật toán F_IFT sẽ được so sánh với các thuật toán của A. Tan [36, 15] và Thang [113]. Thuật toán FW_IFT sẽ được so sánh với thuật toán FW_IFD [113].

3.4.3.1. Kế hoạch thực nghiệm

Các thuật toán được cài đặt bằng ngôn ngữ lập trình Pyhton và chạy trên nền hệ điều hành Window 10 với cấu hình phần cứng là bộ xử lý Core I5, ram 8GB. Cùng với 12 tập dữ liệu thử nghiêm được tải về từ UCI được mô tả chi tiết trong Bảng 3.1. Trong đó $|U|$ là số lượng mẫu, $|C|$ là số thuộc tính điều kiện và $|D|$ là số phân lớp của

thuộc tính quyết định trong mỗi tập dữ liệu (dataset).

Các tập dữ liệu thử nghiệm đều là các dataset có thuộc tính điều kiện miền giá trị số liên tục. Do đó, trước khi thực hiện thuật toán rút gọn thuộc tính, miền giá trị của các thuộc tính sẽ được chuẩn hóa về đoạn [0,1]. Độ tương tự và độ không tương tự của quan hệ ưu tiên mờ trực cảm được tính giống phần thực nghiệm của chương 2 luận án.

Để đánh giá độ chính xác phân lớp của tập rút gọn. Chương này sử dụng hai mô hình phân lớp dữ liệu số là SVM và k-NN(k=|D|). Độ đo đánh giá và phương pháp đánh giá độ chính xác trên các mô hình là độ đo *Accuracy* và phương pháp đánh giá chéo *10-folde* được sử dụng chung cho toàn bộ các tập rút gọn thu được từ các thuật toán.

Bảng 3.1: *Mô tả dữ liệu thực nghiệm*

STT	Tập dữ liệu	Mô tả	$\|U\|$	$\|C\|$	$\|D\|$
1	Wine	Wine	178	13	3
2	Heart	Statlog (Heart)	270	13	2
3	Wdbc	Breast Cancer Wisconsin (Diagnostic)	569	30	2
4	Wpbc	Breast Cancer Wisconsin (Prognostic)	198	33	2
5	Iono	Ionosphere	351	34	2
6	UFDC	Ultrasonic flowmeter diagnostics (C)	181	43	4
7	Sona	Connectionist Bench	208	60	2
8	Libras	Libras Movement	360	90	15
9	Musk	Musk	476	166	2
10	LVB	Voice Rehabilitation(Binary)	126	310	2
11	LVG	Voice Rehabilitation(Gender)	126	310	2
12	PD	Parkinson's Disease Classification	756	754	2

3.4.3.2. Kịch bản thực nghiệm

Nhằm khẳng định phương pháp rút gọn thuộc tính đề xuất là hiệu quả hơn về độ chính xác phân lớp so với một số phương pháp rút gọn thuộc tính khác trên nền tập mờ trực cảm, chương này tiến hành thực nghiệm các thuật toán đề xuất theo các kịch bản như sau:

1) So sánh tập rút gọn của thuật toán đề xuất F_IFT với các thuật toán filter theo

Hình 3.1: *Tập rút gọn thu được từ thuật toán F_IFT*

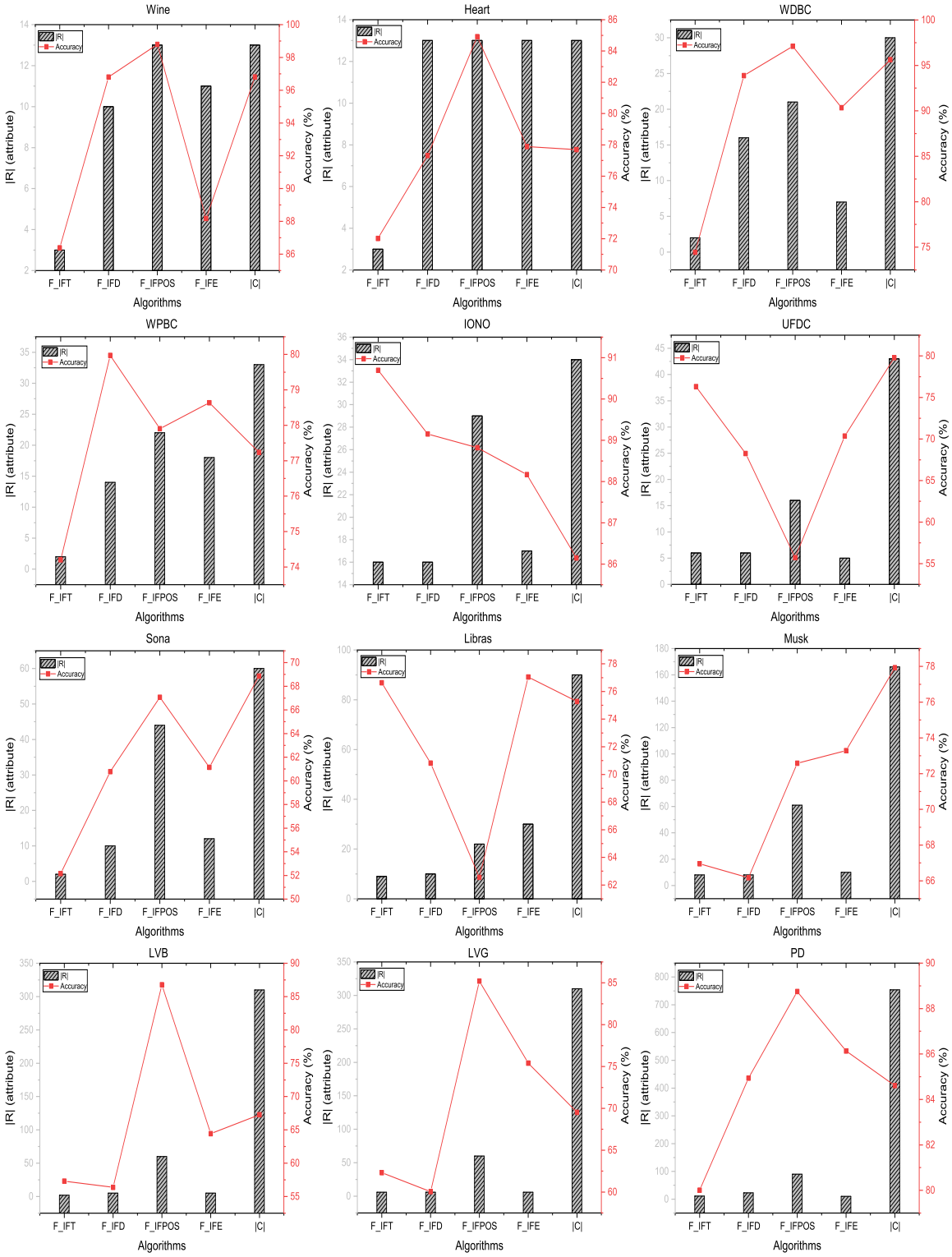
STT	Tập dữ liệu	Tập rút gọn thu được
1	Wine	[7, 11, 9]
2	Heart	[0, 7, 9]
3	Wdbc	[21, 4]
4	Wpbc	[0, 1]
5	Iono	[18, 9, 3, 25, 30, 6, 7, 5, 2, 19, 23, 32, 12, 4, 27, 8]
6	UFDC	[3, 26, 10, 42, 6, 29]
7	Sona	[19, 30]
8	Libras	[66, 1, 71, 0, 88, 39, 89, 51, 58]
9	Musk	[31, 62, 162, 164, 89, 25, 115, 106]
10	LVB	[58, 51]
11	LVG	[79, 82, 56, 84, 47, 49]
12	PD	[0, 420, 421, 422, 423, 424, 426, 427, 430, 437, 502, 613]

tiếp cận khoảng cách mờ trực cảm F_IFD [113], thuật toán filter theo tiếp cận miền dương mờ trực cảm F_IFPOS [36] và thuật toán filter theo tiếp cận Entropy thông tin mờ trực cảm F_IFE [15]. Trong đó các tiêu chí được sử dụng để so sánh và đánh giá bao gồm độ chính xác phân lớp (accuracy), kích thước của tập rút gọn ($|R|$) và thời gian thực hiện của thuật toán (second).

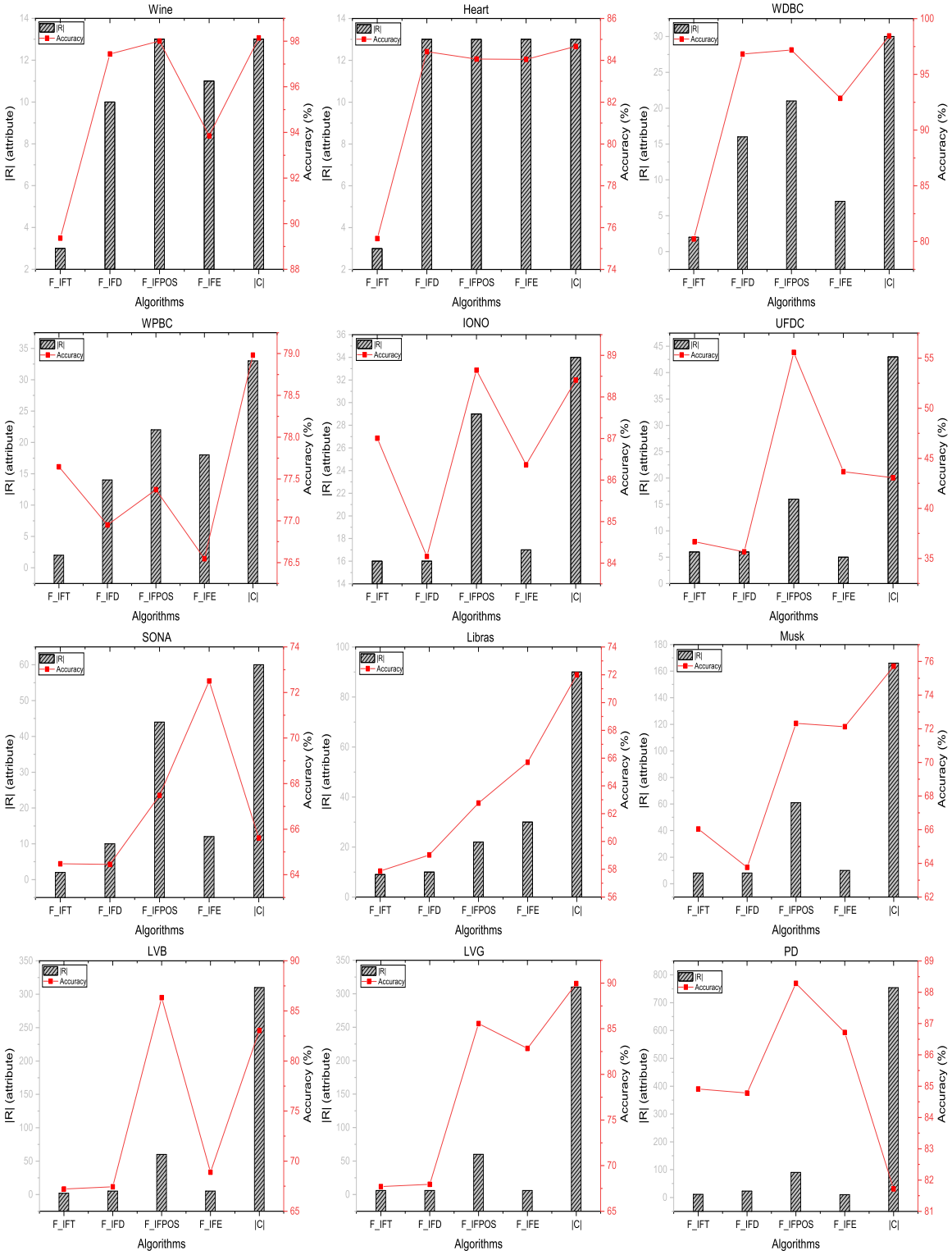
2) So sánh tập rút gọn của thuật toán đề xuất F_IFT với thuật toán filter - wrapper theo tiếp cận khoảng cách mờ trực cảm FW_IFD [113]. Trong đó các tiêu chí được sử dụng để so sánh và đánh giá bao gồm độ chính xác phân lớp (accuracy), kích thước của tập rút gọn ($|R|$) và thời gian thực hiện của thuật toán (second).

3.4.3.3. Đánh giá thuật toán F_IFT

Bảng 3.1 mô tả các tập rút gọn thu được từ thuật toán F_IFT trên từng tập dữ liệu. Trong đó tên của các thuộc tính được đánh số lần lượt từ 0 đến $|C-1|$. Bảng 3.2 so sánh kích thước tập rút gọn thu được từ các thuật toán. Kết quả thực nghiệm được trình bày trong Bảng 3.2 cho thấy kích thước trung bình của các tập rút gọn thu được từ thuật toán đề xuất F_IFT thấp hơn đáng kể so với các thuật toán khác. Quan sát



Hình 3.2: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán so với F_IFT trên mô hình phân lớp KNN.*



Hình 3.3: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán so với F_IFT trên mô hình phân lớp SVM.*

Bảng 3.2: *So sánh kích thước của các tập rút gọn thu được từ các thuật toán theo tiếp cận filter*

STT	F_IFT	F_IFD	F_IFPOS	F_IFE	$\|C\|$
1	3	10	13	11	13
2	3	13	13	13	13
3	2	16	21	7	30
4	2	14	22	18	33
5	16	16	29	17	34
6	6	6	16	5	43
7	2	10	44	12	60
8	9	10	22	30	90
9	8	8	61	10	166
10	2	5	60	5	310
11	6	6	60	6	310
12	5	23	90	10	754

Bảng 3.3: *So sánh độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán theo tiếp cận filter trên mô hình phân lớp KNN*

STT	F_IFT	F_IFD	F_IFPOS	F_IFE	$\|C\|$
1	86.62	96.93	98.96	88.12	96.24
2	72.84	77.57	84.85	77.71	77.7
3	74.91	93.66	97.72	90.48	95.13
4	74.82	79.21	77.61	79	77.81
5	90.89	89.72	88.28	88.63	86.26
6	76.71	68.9	55.65	70.61	79.2
7	52.83	60.67	67.45	61.93	68.95
8	76.62	70.79	62.46	77.88	75.21
9	66.14	66.17	72.25	73.16	77.61
10	57.55	56.31	86.76	64.92	67.85
11	62.07	60.91	85.83	75.76	69.54
12	80.11	84.91	88.28	86.6	84.05

biểu đồ phân tích mối quan hệ về kích thước tập rút gọn (trái) trong Hình 3.4, ta có thể thấy số thuộc tính tăng lên nhưng kích thước tập rút gọn không tăng. Do đó, chúng ta có thể khẳng định phương pháp rút gọn thuộc tính theo tiếp cận tôpô giảm chiều tốt hơn so với các tiếp cận đo đo trên nền tập mờ trực cảm.

Hơn nữa, thời gian thực hiện của thuật toán rút gọn thuộc tính đề xuất F_IFT cũng

Bảng 3.4: *So sánh độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán theo tiếp cận filter trên mô hình phân lớp SVM*

STT	F_IFT	F_IFD	F_IFPOS	F_IFE	$\|C\|$
1	89.37	97.45	98.14	93.55	98.64
2	75.33	84.35	84.81	84.36	84.91
3	80.79	96.96	97.62	92.34	98.99
4	77.71	76.3	77.51	76.13	78.9
5	87.09	84.46	88.17	86.22	88.07
6	36.39	35.13	55.7	43.94	43.92
7	64.23	64.34	67.18	72.71	65.3
8	57.46	59.31	62.53	65.92	71.68
9	66.48	63.76	72.22	72.71	75.49
10	67.03	67.55	86.45	68.86	83.06
11	67.83	67.98	85.3	82.37	89.92
12	84.83	84.75	88.38	86.39	81.37

được cải thiện đáng kể so với các thuật toán khác. Quan sát biểu đồ phân tích mối quan hệ về thời gian thực hiện (phải) trong Hình 3.4, ta có thể thấy thời gian thực hiện của thuật toán hầu như không biến động trên các bộ dữ liệu có số lượng thuộc tính nhỏ hơn 500, và biến động thấp nhất trên các bộ dữ liệu có số lượng thuộc tính lớn hơn 500.

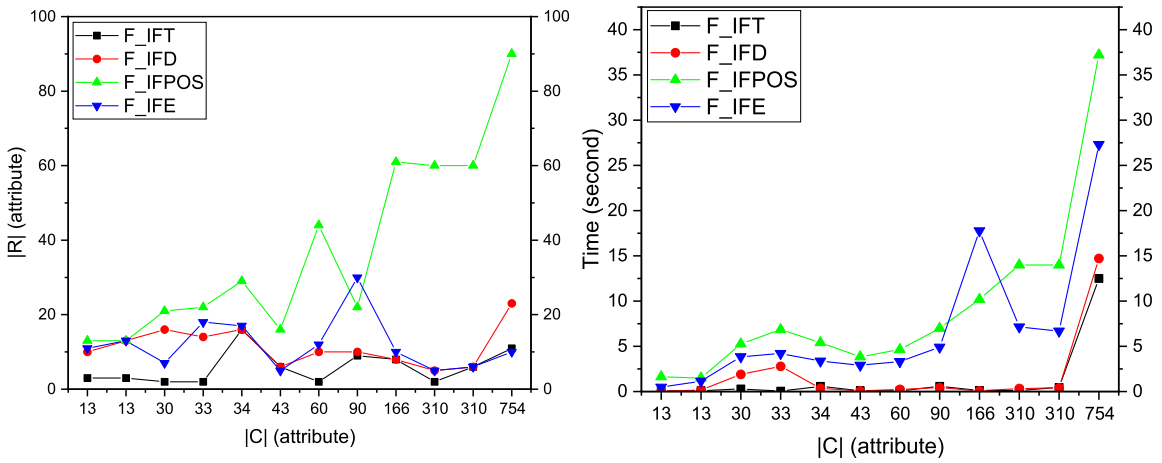
Tuy nhiên, độ chính xác phân lớp của tập rút gọn thu được từ thuật toán đề xuất vẫn còn hạn chế. Quan sát kết quả thống kê về độ chính xác phân lớp các tập rút gọn thu được từ thuật toán đề xuất trên mô hình phân lớp KNN trong Bảng 3.3, trên mô hình phân lớp SVM trong Bảng 3.4. Ta có thể thấy độ chính xác trung bình trên toàn bộ các tập dữ liệu của hai mô hình đều thấp hơn so với các thuật toán khác. Sau đây là các phân tích về nguyên nhân ảnh hưởng tới thời gian thực hiện của thuật toán F_IFT, kích thước và độ chính xác phân lớp của tập rút gọn thu được bởi thuật toán F_IFT.

- *Kích thước của tập rút gọn*: Các kết quả phân tích và thống kê trong phần thực nghiệm cho thấy thuật toán đề xuất F_IFT hiệu quả về giảm thuộc tính. Nguyên nhân chính ảnh hưởng tốt tơi khả năng giảm thuộc tính đó là phương pháp định nghĩa tập rút gọn theo tiếp cận tôpô như đã được đề xuất trong phần nghiên cứu lý thuyết của

Chương 3. Kết quả này là hoàn toàn phù hợp với phương pháp đánh giá độ tương đồng về mặt cấu trúc trong tôpô như đã được khẳng định trong các kết quả nghiên cứu của Yu và các cộng sự trong [39].

- *Thời gian thực hiện của thuật toán*: Cũng dựa trên phương pháp định nghĩa tập rút gọn theo tiếp cận tôpô. Thay vì phải đối sánh độ tương đồng giữa tôpô của tập thuộc rút gọn thông qua B_R với tôpô của tập thuộc tính ban đầu thông qua B_C, chương này sử dụng khái niệm tôpô đơn vị thông qua B_I để làm điều kiện dừng cho thuật toán. Khi đó, chúng ta không phải tính toán B_C nên giảm đáng kể thời gian thực hiện của thuật toán. Hơn nữa khi tập rút gọn có kích thước càng nhỏ thì thời gian hội tụ của thuật toán càng nhanh. Đây là những nguyên nhân quan trọng cải thiện tốt thời gian thực hiện của thuật toán F_IFT.

- *Độ chính xác phân lớp*: Trái lại với kích thước của tập rút gọn được cải thiện đáng kể thì độ chính xác phân lớp còn gặp nhiều hạn chế. Nguyên nhân chính đó là phương pháp đánh giá tập rút gọn đề xuất trong nghiên cứu của Chương 3 vẫn dựa trên tiếp cận độ đo độ tương tự. Đây cũng chính là nhược điểm của hầu hết các phương pháp rút gọn thuộc tính hiện nay như đã được trình bày trong phần Mở đầu của luận án.



Hình 3.4: *Biểu đồ đánh giá mối quan hệ về kích thước tập rút gọn (trái) và thời gian thực hiện (phải) với số lượng thuộc tính ban đầu của thuật toán F_IFT so với các thuật toán khác*

3.4.3.4. Đánh giá thuật toán FW_IFT

Các Bảng 3.5 và Bảng 3.6 trình bày tập rút gọn thu được từ thuật toán đề xuất FW_IFT trên từng tập dữ liệu tương ứng trên hai mô hình phân lớp SVM và KNN. Trong các bảng này, tên của các thuộc tính cũng được đánh số thứ tự từ 0 đến $|C|-1$ để tiện cho quá trình thống kê và quan sát các thuộc tính thu được sau rút gọn.

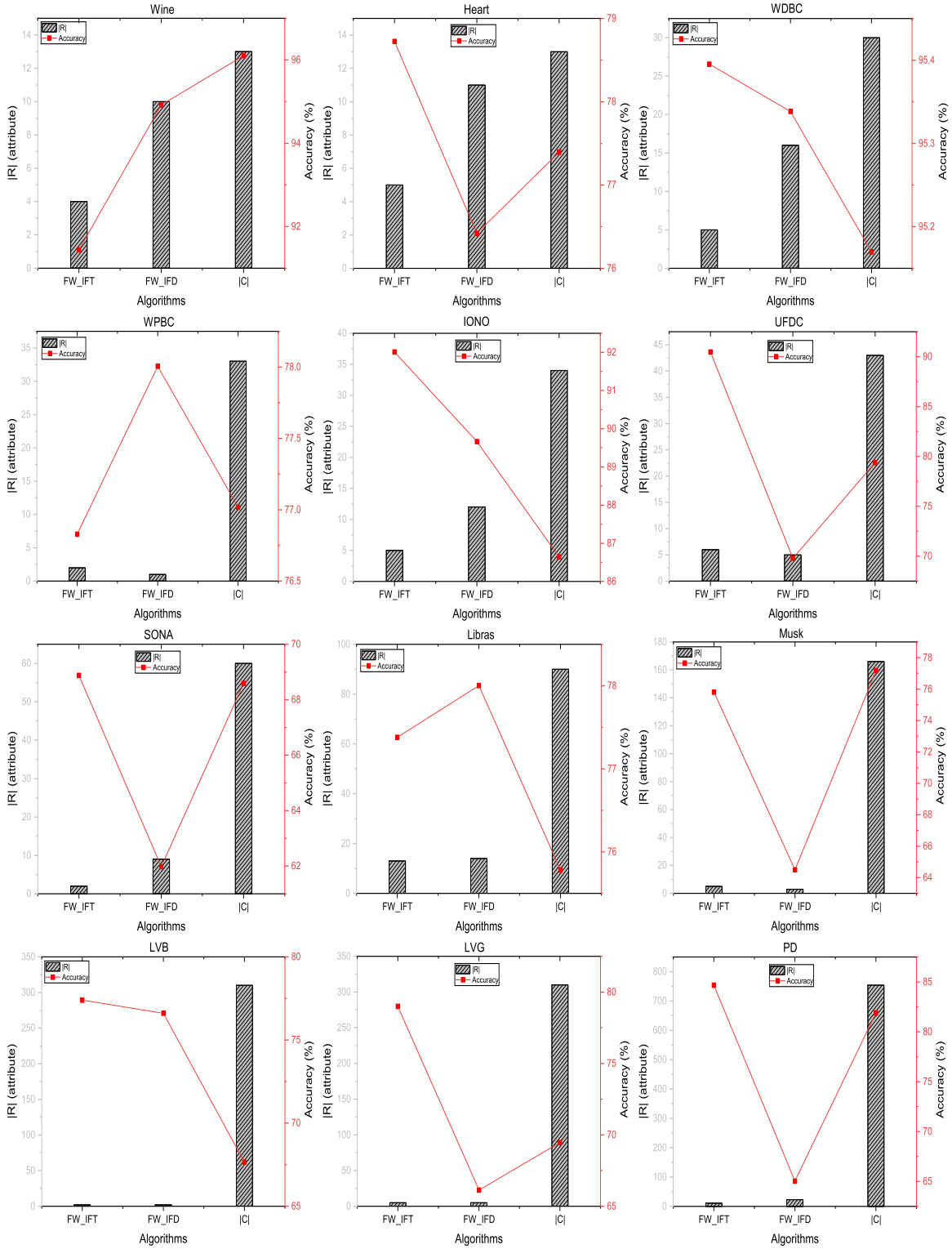
Bảng 3.7 so sánh kích thước tập rút gọn thu được từ các thuật toán. Cả hai thuật toán so sánh đều sử dụng phương pháp lai ghép filter - wrapper trên hai mô hình phân lớp dữ liệu SVM và KNN.

Bảng 3.5: *Tập rút gọn thu được từ thuật toán FW_IFT trên mô hình phân lớp SVM*

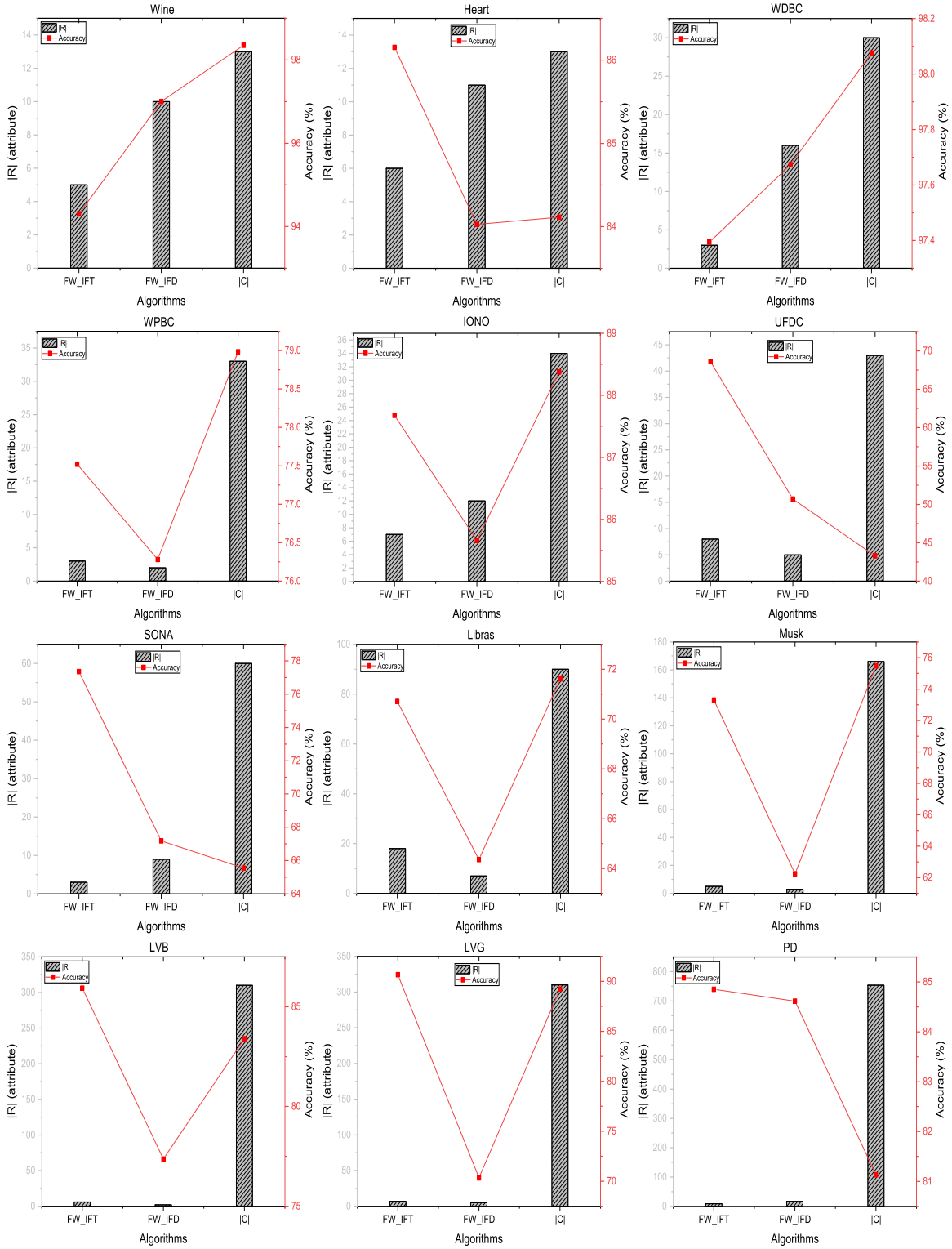
STT	Tập dữ liệu	Tập rút gọn
1	Wine	[7, 9, 10, 11, 12]
2	Heart	[0, 8, 9, 11, 12]
3	Wdbc	[21, 22, 24]
4	Wpbc	[0, 1]
5	Iono	[18, 0, 2, 3, 4, 5, 10]
6	UFDC	[3, 17, 19, 20, 21, 22, 23, 25]
7	Sona	[19, 10, 11]
8	Libras	[66, 10, 11, 14, 15, 16, 19, 20, 23, 26, 29, 34, 35, 41, 44, 51, 61, 72]
9	Musk	[31, 161, 162, 163, 164]
10	LVB	[58, 81, 82, 83, 84, 88]
11	LVG	[79, 88, 89, 90, 91, 92, 93]
12	PD	[0, 420, 421, 422, 427, 430, 437, 502, 613]

Kết quả thực nghiệm được trình bày trong Bảng 3.7 cho thấy kích thước trung bình của các tập rút gọn thu được từ thuật toán đề xuất FW_IFT cải thiện tốt hơn so với thuật toán theo tiếp cận độ đo khoảng cách mờ. Quan sát biểu đồ phân tích mối quan hệ về kích thước tập rút gọn (trái) trong Hình 3.7 và Hình 3.8, ta có thể thấy số thuộc tính tăng lên nhưng kích thước tập rút gọn không tăng đáng kể, hơn nữa tính ổn định về kích thước đạt được trên hầu hết các bộ dữ liệu.

Hơn nữa, độ chính xác phân lớp của các tập rút gọn thu được từ thuật toán đề xuất được cải thiện đáng kể, tốt hơn so với thuật toán theo tiếp cân độ đo khoảng cách mờ



Hình 3.5: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán so với FW_IFT trên mô hình phân lớp KNN.*



Hình 3.6: *Mối quan hệ về kích thước và độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán so với FW_IFT trên mô hình phân lớp SVM.*

Bảng 3.6: *Tập rút gọn thu được từ thuật toán FW_IFT trên mô hình phân lớp KNN*

STT	Tập dữ liệu	Tập rút gọn
1	Wine	[7, 10, 11, 12]
2	Heart	[0, 8, 9, 11, 12]
3	Wdbc	[19, 20, 21, 24, 28]
4	Wpbc	[0, 1]
5	Iono	[18, 3, 4, 5, 9]
6	UFDC	[3, 29, 31, 33, 35, 36]
7	Sona	[19, 11]
8	Libras	[66, 4, 12, 13, 16, 19, 22, 25, 33, 36, 43, 54, 59]
9	Musk	[31, 109, 110, 111, 114]
10	LVB	[58, 86]
11	LVG	[79, 90, 91, 92, 93]
12	PD	[0, 420, 421, 422, 427, 430, 437, 502, 503, 504, 613]

trực cảm. Các kết quả phân tích và thống kê về độ chính xác phân lớp của tập rút gọn thu từ thuật toán FW_IFT trên cả hai mô hình phân lớp KNN và SVM trong Bảng 3.8 cho thấy độ chính xác phân lớp trung bình trên cả hai mô hình phân lớp đều tốt hơn đáng kể so với phương pháp dựa trên khoảng cách mờ trực cảm. Đặc biệt, một số bộ dữ liệu có độ chính xác phân lớp ban đầu thấp như Sona và UFDC đã được cải thiện đáng kể so với thuật toán được so sánh.

Tuy nhiên thời gian thực hiện của thuật toán đề xuất FW_IFT còn gặp nhiều hạn chế. Đây là sự đánh đổi thời gian thực hiện để cải thiện độ chính xác phân lớp cho thuật toán F_IFT. Quan sát biểu đồ phân tích mối quan hệ về thời gian thực hiện của các thuật toán (phải) trong Hình 3.7 và Hình 3.8, ta có thể thấy số thuộc tính tăng lên thì thời gian thực hiện của thuật toán đề xuất cũng tăng lên đáng kể trên hầu hết các bộ dữ liệu. Sau đây là các phân tích về nguyên nhân ảnh hưởng tới thời gian thực hiện của thuật toán FW_IFT, kích thước và độ chính xác phân lớp của tập rút gọn thu được bởi thuật toán FW_IFT.

- *Kích thước của tập rút gọn*: Các kết quả phân tích được trình bày trong phần thực nghiệm của Chương 3 trong luận án cho thấy kích thước các tập rút gọn thu được từ thuật toán FW_IFT gần như tương đương với thuật toán F_IFT. Theo cùng tiếp cận

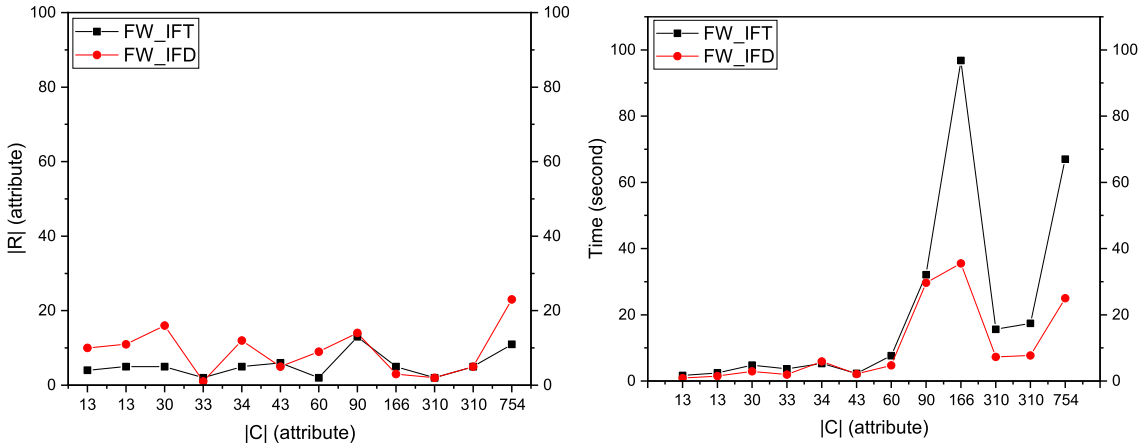
Bảng 3.7: *So sánh kích thước của các tập rút gọn thu được từ các thuật toán theo tiếp cận filter - wrapper trên mô hình phân lớp SVM và KNN*

STT	Tập dữ liệu	FW_IFT		FW_IFD		$\|C\|$
		SVM	KNN	SVM	KNN	
1	Wine	5	4	10	10	13
2	Heart	6	5	11	11	13
3	Wdbc	3	5	16	16	30
4	Wpbc	3	2	2	2	33
5	Iono	7	5	12	12	34
6	UFDC	8	6	5	5	43
7	Sona	3	2	9	9	60
8	Libras	18	13	7	14	90
9	Musk	5	5	3	3	166
10	LVB	6	2	2	2	310
11	LVG	7	5	5	5	310
12	PD	9	11	17	23	754

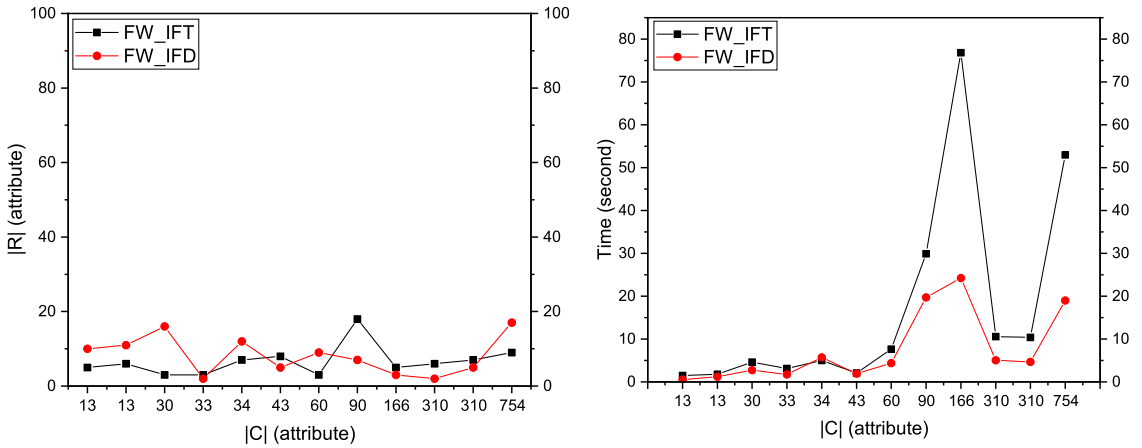
Bảng 3.8: *So sánh độ chính xác phân lớp của các tập rút gọn thu được từ các thuật toán theo tiếp cận filter - wrapper trên mô hình phân lớp SVM và KNN*

STT	Tập dữ liệu	FW_IFT		FW_IFD		$\|C\|$	
		SVM	KNN	SVM	KNN	SVM	KNN
1	Wine	94.24	91.25	97.87	94.74	98.16	96.25
2	Heart	86.43	78.85	84.65	76.74	84.5	77.44
3	Wdbc	97.15	95.42	97.99	95.02	98.33	95.45
4	Wpbc	77.79	76.12	76.14	78.34	78.02	77.18
5	Iono	87.1	92.05	85.46	89.14	88.37	86.04
6	**UFDC**	68.16	90.9	50.95	69.14	43.49	79.13
7	**Sona**	77.21	68.35	67.35	61	65.45	68.16
8	Libras	70.9	77.59	64.79	78.02	71.41	75.23
9	Musk	73.17	75.13	62.51	64.41	75.54	77.37
10	LVB	85.29	77.19	77.71	76.31	83.24	67.8
11	LVG	90.22	78.64	70.18	66.93	89.05	69.22
12	PD	84.47	84.79	84.8	65.53	81.26	81.8

lai ghép filter - wrapper, thuật toán đề xuất theo tiếp cận tôpô mờ trực cảm cho tập rút gọn có kích thước tốt hơn thuật toán theo tiếp cận khoảng cách mờ trực cảm. Yếu tố ảnh hưởng chính vẫn là phương pháp định nghĩa tập rút gọn theo tiếp cận tôpô như đã được trình bày.



Hình 3.7: *Biểu đồ đánh giá mối quan hệ về kích thước tập rút gọn (trái) và thời gian thực hiện (phải) với số lượng thuộc tính ban đầu của thuật toán FW_IFT so với các thuật toán khác trên mô hình phân lớp KNN*



Hình 3.8: *Biểu đồ đánh giá mối quan hệ về kích thước tập rút gọn (trái) và thời gian thực hiện (phải) với số lượng thuộc tính ban đầu của thuật toán FW_IFT so với các thuật toán khác trên mô hình phân lớp SVM*

- *Độ chính xác phân lớp của tập rút gọn*: Yếu tố ảnh hưởng đến khả năng cải thiện độ chính xác phân lớp của thuật toán đó chính là phương pháp filter - wrapper thông qua cấu trúc dữ liệu Stack. Nhược điểm của tiếp cận độ đo là bỏ sót nhiều thuộc tính ứng viên có cùng độ quan trọng, do đó cấu trúc Stack sẽ lưu vết lại các tập rút gọn ứng viên này để xây dựng các tập rút gọn ứng viên. phục vụ cho bước wrapper của thuật toán.

- *Thời gian thực hiện của thuật toán*: Trái lại với khả năng nâng cao độ chính

xác của tập rút gọn thu được từ thuật toán thì thời gian thực hiện còn nhiều hạn chế. Nguyên nhân chính cũng là phương pháp xây dựng các tập rút gọn ứng viên thông qua cấu trúc dữ liệu Stack. Theo tiếp cận này ta sẽ thu được nhiều tập rút gọn ứng viên để xét nhưng phải trả giá về mặt thời gian xác định tập rút gọn cuối cùng của thuật toán.

3.5. Kết luận Chương 3

Chương 3, luận án trình bày về phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm. Các đóng góp chính của Chương này gồm có:

- Đề xuất cấu trúc tôpô mờ trực cảm dựa trên quan hệ ưu tiên mờ trực cảm. Nghiên cứu các tính chất của IF-base và IF-subbase và các phép toán cơ bản nhằm xây dựng độ đo đánh giá sự tương đồng giữa hai tôpô mờ trực cảm.

- Đề xuất hai thuật toán theo phương pháp filter và filter - wrapper tìm tập rút gọn trong bảng quyết định số với định nghĩa mới về tập rút gọn theo tiếp cận tôpô đơn vị.

Các kết quả thực nghiệm cho thấy thuật toán đề xuất theo phương pháp filter cho thời gian thực hiện hiệu quả. Trong khi đó thuật toán đề xuất theo phương pháp lai ghép filter - wrapper cho các tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp trên hầu hết các tập dữ liệu, đặc biệt là mục tiêu nâng cao chất lượng phân lớp và giảm kích thước của tập rút gọn cho các bộ dữ liệu có độ chính xác phân lớp ban đầu thấp.

CHƯƠNG 4. PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRONG BẢNG QUYẾT ĐỊNH THEO TIẾP CẬN TÔPÔ HAUSDORFF

4.1. Mở đầu

Năm 2005, Lashin và các cộng sự lần đầu tiên giới thiệu khái niệm tôpô rút gọn theo tiếp cận rough set [38]. Từ đó, phương pháp xây dựng cấu trúc tôpô theo tiếp cận tập thô được nhiều nhà nghiên cứu quan tâm và đề xuất. Hiện nay có hai phương pháp xây dựng tôpô theo tiếp cận tập thô gồm có, các phương pháp xây dựng tôpô từ không gian xấp xỉ của tập thô [41, 42, 43, 39], các phương pháp xây dựng tôpô từ các phép toán xấp xỉ của tập thô [44].

Mặc dù phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm cho tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp. Tuy nhiên thời gian thực hiện còn chưa hiệu quả trên các bộ dữ liệu có số chiều và mẫu lớn. Bên cạnh đó, phương pháp chọn lọc thuộc tính của các phương pháp đề xuất hiện nay vẫn theo tiếp cận xây dựng các độ đo, khó khả thi trên các tập dữ liệu lớn (big data). Hơn nữa, với các ưu điểm của cấu trúc tôpô Hausdorff, chương này đề xuất phương pháp rút gọn thuộc tính theo tiếp cận tôpô Hausdorff với các bước chính như sau:

- Xây dựng cấu trúc tôpô theo tiếp cận tập thô trên không gian xấp xỉ mờ trực cảm ngưỡng β.

- Đề xuất phương pháp xác định cấu trúc tôpô Hausdorff của thuộc tính, làm cơ sở để xây dựng phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm đồng cấu trúc thuộc tính, làm cơ sở để phân nhóm các thuộc tính có cấu trúc tôpô Hausdorff.

Trên cơ sở đó, luận án đề xuất phương pháp wrapper các nhóm thuộc tính cho tập

rút gọn hiệu quả về thời gian và độ chính xác phân lớp.

Các kết quả nghiên cứu trong Chương này được công bố trên các công trình nghiên cứu [CT1]. [CT5] đang chờ phản biện vòng 1.

4.2. Đề xuất cấu trúc tôpô từ không gian xấp xỉ mờ ngưỡng β

Định nghĩa 4.1 (Không gian xấp xỉ mờ ngưỡng β)**.** Không gian xấp xỉ mờ ngưỡng β kí hiệu bởi (U, R^β). Trong đó $R^\beta = \{R(p,q) \,|\beta \leq R(p,q), \beta \in [0,1], \forall p,q \in U\}$.

Định nghĩa 4.2 (Công thức quan hệ mờ ngưỡng β)**.** Quan hệ tương đương mờ ngưỡng β của $p,q \in U$ được định nghĩa như sau:

$$R^\beta(p,q) = \begin{cases} 1-|p-q| : \text{nếu } 1-|p-q| \geq \beta \\ 0 : \text{nếu } 1-|p-q| < \beta. \end{cases} \tag{4.1}$$

Định nghĩa 4.3 (Thứ tự bộ phận của hai quan hệ mờ ngưỡng β)**.** Cho hai quan hệ tương đương mờ R_1^β and R_2^β xác định trên U. Khi đó R_1^β được gọi là nhỏ hơn $(\prec)$ R_2^β nếu mọi $p,q \in U$ ta có $R_1^\beta(p,q) \leq R_2^\beta(p,q)$.

Mệnh đề 4.1 (Cấu trúc tôpô theo tiếp cận tập thô)**.** *Cho không gian xấp xỉ (U, R^β) và R^β là một quan hệ tương đương mờ. Khi đó $\mathscr{T} = \left\{X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X)\right\}$ là một tôpô xác định trên U.*

Chứng minh. Ta cần chứng minh 3 điều kiện ràng buộc của cấu trúc tôpô như sau:

(1): Dựa trên tính chất 2) của mệnh đề 1.1, ta có $\underline{R^\beta}(U) = U$ và $\overline{R^\beta}(\emptyset) = \emptyset$. Khi đó $\emptyset \in \mathscr{T}$ và $U \in \mathscr{T}$;

(2): Giả sử $X, Y \in \mathscr{T}$, khi đó $\underline{R^\beta}(X) = \overline{R^\beta}(X)$, và $\underline{R^\beta}(Y) = \overline{R^\beta}(Y)$. Hơn nữa $\underline{R^\beta}(X \cap Y) \subseteq \overline{R^\beta}(X \cap Y)$ và $\overline{R^\beta}(X \cap Y) \subseteq \overline{R^\beta}(X) \cap \overline{R^\beta}(Y) = \underline{R^\beta}(X) \cap \underline{R^\beta}(Y) = \underline{R^\beta}(X \cap Y)$. Khi đó $\overline{R^\beta}(X \cap Y) = \underline{R^\beta}(X \cap Y)$. Do đó $X \cap Y \in \mathscr{T}$;

(3): Giả sử $X_k \in \mathscr{T} | k \in K$. Khi đó $\underline{R^\beta}(X_k) = \overline{R^\beta}(X_k)$, do đó với mọi $k \in K$ ta có $\overline{R^\beta}(\bigcup_{k\in K} X_k) = \bigcup_{k\in K} \overline{R^\beta}(X_k) = \bigcup_{k\in K} \underline{R^\beta}(X_k) \subseteq \underline{R^\beta}(\bigcup_{k\in K} X_k)$. Hơn nữa $\underline{R^\beta}(\bigcup_{k\in K} X_k) \subseteq \overline{R^\beta}(\bigcup_{k\in K} X_k)$ do đó $\underline{R^\beta}(\bigcup_{k\in K} X_k) = \overline{R^\beta}(\bigcup_{k\in K} X_k)$.

Từ (1), (2) và (3) ta có thể kết luận $\mathscr{T}$ là một tôpô trên U. □

Ví dụ 4.1. Xét bảng quyết định như trong Bảng 1.3, sử dụng công thức quan hệ 4.1 với $\beta = 0.5$ ta có:

$$R_a^\beta = \begin{bmatrix} 1 & 1 & 0.8 & 0 & 0 & 0 \\ 1 & 1 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0.8 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}, R_c^\beta = \begin{bmatrix} 1 & 0 & 0.8 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0.8 & 0 & 1 & 0.8 & 0.8 & 0 \\ 1 & 0 & 0.8 & 1 & 1 & 0 \\ 1 & 0 & 0.8 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Khi đó:

$$\mathscr{T}_a = \{\emptyset, \{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, U\}; \mathscr{T}_c = \{\emptyset, \{u_1, u_3, u_4, u_5\}, \{u_2, u_6\}, U\}$$

Định nghĩa 4.4 (Quan hệ bao thuộc của hai tôpô)**.** Cho $\mathscr{T}_p = \left\{ X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X) \right\}$ và $\mathscr{T}_q = \left\{ X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X) \right\}$ là hai tôpô xác định trên U tương ứng với $p, q \subseteq C$. Khi đó $\mathscr{T}_p \subseteq \mathscr{T}_q$ nếu với mọi $e \in \mathscr{T}_p$ thì $e \in \mathscr{T}_q$.

Ví dụ 4.2. Cho hai quan hệ tương đương mờ như sau:

$$R_p^\beta = \begin{bmatrix} 1 & 1 & 0.8 & 0 & 0 & 0 \\ 1 & 1 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0.8 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}, R_q^\beta = \begin{bmatrix} 1 & 0 & 0.8 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0.8 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$\mathscr{T}_p = \{\emptyset, \{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, U\};$$

Khi đó: $\mathscr{T}_q = \{\emptyset, \{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, \{u_1, u_3, u_4, u_5\}, \{u_2, u_6\}, \{u_2\},$.

$$\{u_1, u_3, u_4, u_5, u_6\}, \{u_6\}, \{u_1, u_2, u_3, u_4, u_5\}, \{u_4, u_5\}, \{u_1, u_2, u_3, u_6\}, U\}$$

Do đó: $\mathscr{T}_p \subseteq \mathscr{T}_q$

Mệnh đề 4.2 (Thứ tự bộ phận của hai tôpô)**.** *Cho* $\mathscr{T}_p = \left\{ X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X) \right\}$ *và*

$\mathscr{T}_q = \left\{ X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X) \right\}$ *là hai tôpô xác định trên* U *tương ứng với* $p,q \subseteq C$. *Khi đó* $\mathscr{T}_p \subseteq \mathscr{T}_q$ *nếu* $R_q^\beta \prec R_p^\beta$.

Chứng minh. Giả sử rằng $R_p^\beta \prec R_q^\beta$, khi đó với mọi $u \in U$, ta có $[u]_p^\beta \subseteq [u]_q^\beta$. Do đó $u \subseteq U$, nếu $[u]_q^\beta \subseteq X$ thì $[x]_p^\beta \subseteq X$. Ta có đpcm. □

Ví dụ 4.3. Quan sát ví dụ 4.2, chúng ta có thể thấy rõ vì $R_p^\beta \prec R_q^\beta$ nên $\mathscr{T}_p \leq \mathscr{T}_q$.

Định nghĩa 4.5 (Quan hệ mịn nhất)**.** Cho R_1^β là một quan hệ tương đương mờ trên U, khi đó R_1^β được gọi là mịn nhất khi và chỉ khi với mọi $p,q \in U$, $R_1^\beta(p,q) = 1$ nếu $p = q$ và $R_1^\beta(p,q) = 0$ nếu $p \neq q$.

Mệnh đề 4.3 (Tôpô lớn nhất)**.** *Cho* $\mathscr{T}_1 = \left\{ X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X) \right\}$. *Khi đó* $\mathscr{T}_1$ *được gọi là lớn nhất nếu* $R^\beta = R_1^\beta$.

Chứng minh. Chứng minh tương tự như mệnh đề 4.2 ta có đpcm. □

Ví dụ 4.4. Cho ma trận quan hệ mờ: $R_1^\beta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

Khi đó: $\mathscr{T}_1 = \left\{ X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X) \right\} = P(U) = 2^U$

Định nghĩa 4.6 (Quan hệ thô nhất)**.** Cho R_0^β là một quan hệ tương đương mờ trên U, khi đó R_0^β được gọi là thô nhất khi và chỉ khi với mọi $p,q \in U$, $R_0^\beta(p,q) = 1$.

Mệnh đề 4.4 (Tôpô nhỏ nhất)**.** *Cho* $\mathscr{T}_0 = \left\{ X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X) \right\}$. *Khi đó* $\mathscr{T}_0$ *được gọi là nhỏ nhất nếu* $R^\beta = R_0^\beta$.

Chứng minh. Chứng minh tương tự như mệnh đề 4.2 ta có đpcm. □

Ví dụ 4.5. Cho ma trận quan hệ mờ: $R_0^\beta = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$.

Khi đó: $\mathscr{T}_0 = \left\{X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X)\right\} = \{\emptyset, U\}$

Định nghĩa 4.7 (Quan hệ bù)**.** Cho quan hệ tương đương R^β xác định trên U. Khi đó quan hệ bù của quan hệ R^β được định nghĩa như sau: với mọi $p, q \in U$

$$\left(R^\beta\right)^c (p,q) = \begin{cases} 1 - R^\beta(p,q) \,|\, 1 - R^\beta(p,q) \geq \beta \\ 0 | 1 - R^\beta(p,q) < \beta \end{cases} \tag{4.2}$$

Mệnh đề 4.5 (Tôpô bù)**.** *Cho* $\mathscr{T} = \left\{X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X)\right\}$.

Khi đó $\sim \mathscr{T} = \left\{X \subseteq U | \sim \underline{R^\beta}(X) = \sim \overline{R^\beta}(X)\right\}$ *là một tôpô bù của* $\mathscr{T}$.

Ví dụ 4.6. Quan sát các ví dụ 4.4 và 4.5, chúng ta có thể thấy $\mathscr{T}_0$ và $\mathscr{T}_1$ là các tôpô bù của nhau.

Định nghĩa 4.8 (Phép toán hợp hai tôpô)**.** Cho $\mathscr{T}_p = \left\{X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X)\right\}$ và $\mathscr{T}_q = \left\{X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X)\right\}$ là hai tôpô xác định trên U tương ứng với $p, q \subseteq C$. Khi đó hợp của hai tôpô được định nghĩa như sau:

$$\mathscr{T}_p \cup \mathscr{T}_q = \left\{X \subseteq U | X \in \mathscr{T}_p \vee X \in \mathscr{T}_q\right\} \tag{4.3}$$

Mệnh đề 4.6 (Hợp hai tôpô theo tiếp cận tập thô)**.** *Cho* $\mathscr{T}_p = \left\{X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X)\right\}$ *và* $\mathscr{T}_q = \left\{X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X)\right\}$ *là hai tôpô xác định trên U tương ứng với* $p, q \subseteq C$*. Khi đó, nếu* $R_q \prec R_p$ *hoặc* $R_p \prec R_q$ *thì* $\mathscr{T}_p \cup \mathscr{T}_q = \left\{X \subseteq U | \underline{R^\beta}_{pq}(X) = \overline{R^\beta}_{pq}(X)\right\}$*, với* $R_{pq}^\beta = R_p^\beta \cap R_q^\beta$.

Chứng minh. Giả sử rằng $R_q^\beta \prec R_p^\beta$ hoặc $R_p^\beta \prec R_q^\beta$, do đó với mọi $u \in U$, ta có $[u]_p^\beta \subseteq$

$[u]_q^\beta$ hoặc $[u]_q^\beta \subseteq [u]_p^\beta$. Khi đó, với mọi $u \subseteq U$, nếu $[u]_q^\beta \subseteq X$ thì $[u]_p^\beta \subseteq X$ hay nếu $[u]_p^\beta \subseteq X$ thì $[u]_q^\beta \subseteq X$. Khi đó $[u]_p^\beta \cap [u]_q^\beta = [u]_p^\beta$ hoặc $[u]_p^\beta \cap [u]_q^\beta = [u]_q^\beta$. Do đó, ta có đpcm. □

Ví dụ 4.7. Xem ví dụ 4.2 để thêm thông tin chi tiết.

Định nghĩa 4.9 (Phép toán giao hai tôpô)**.** Cho $\mathscr{T}_p = \left\{ X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X) \right\}$ và $\mathscr{T}_q = \left\{ X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X) \right\}$ là hai tôpô xác định trên U tương ứng với $p, q \subseteq C$. Khi đó giao của hai tôpô được định nghĩa như sau:

$$\mathscr{T}_p \cap \mathscr{T}_q = \left\{ X \subseteq U | X \in \mathscr{T}_p \wedge X \in \mathscr{T}_q \right\} \tag{4.4}$$

Mệnh đề 4.7 (Giao hai tôpô theo tiếp cận tập thô)**.** *Cho* $DT = (U, C, D, f)$ *với* $\mathscr{T}_p = \left\{ X \subseteq U | \underline{R_p^\beta}(X) = \overline{R_p^\beta}(X) \right\}$ *và* $\mathscr{T}_q = \left\{ X \subseteq U | \underline{R_q^\beta}(X) = \overline{R_q^\beta}(X) \right\}$ *là hai tôpô xác định trên* U *tương ứng với* $p, q \subseteq C$. *Khi đó* $\mathscr{T}_p \cap \mathscr{T}_q = \left\{ X \subseteq U | \underline{R^\beta}_{pq}(X) = \overline{R^\beta}_{pq}(X) \right\}$ *với* $R_{pq}^\beta = R_p^\beta \cup R_q^\beta$.

Chứng minh. Giả sử rằng $X \in R_p^\beta \cup R_q^\beta$, Khi đó $X \subseteq \underline{R_p^\beta \cup R_q^\beta}(X)$, do đó $X \subseteq \underline{R^\beta}_p \cap \underline{R_p^\beta} \leftrightarrow X \subseteq \underline{R_{pq}^\beta} \leftrightarrow X \in \mathscr{T}_p \cap \mathscr{T}_q$. Ta có đpcm. □

Định nghĩa 4.10 (Nhóm tôpô)**.** Cho không gian xấp xỉ mờ (U, R) và tôpô $\mathscr{T} = \{X \subseteq U | \underline{R}(X) = \overline{R}(X)\}$ xác định trên U. Khi đó họ các tôpô cùng với các phép toán kí hiệu bởi $G\langle (U,R), \cup, \cap, \sim, \mathscr{T}_0, \mathscr{T}_1 \rangle$ được gọi là một nhóm nếu với mọi $\mathscr{T} \in G$ thỏa mãn:

(1): $(\mathscr{T}_a \cup \mathscr{T}_b) \cup \mathscr{T}_c = \mathscr{T}_a \cup (\mathscr{T}_b \cup \mathscr{T}_c)$; (2): $(\mathscr{T}_a \cap \mathscr{T}_b) \cap \mathscr{T}_c = \mathscr{T}_a \cap (\mathscr{T}_b \cap \mathscr{T}_c)$;

(3): $(\mathscr{T}_a \cup \mathscr{T}_b) \cap \mathscr{T}_c = \mathscr{T}_a \cap \mathscr{T}_c \cup \mathscr{T}_b \cap \mathscr{T}_c$; (4): $\mathscr{T}_a \cup \mathscr{T}_b = \mathscr{T}_b \cup \mathscr{T}_a$;

(5): $\mathscr{T}_a \cap \mathscr{T}_b = \mathscr{T}_b \cap \mathscr{T}_a$; (6): $\mathscr{T}_0 \cup \mathscr{T}_a = \mathscr{T}_a$; (7): $\mathscr{T}_1 \cap \mathscr{T}_a = \mathscr{T}_a$.

Mệnh đề 4.8 (Nhóm tôpô abel)**.** *Cho không gian xấp xỉ mờ* (U, R^β) *với tôpô* $\mathscr{T} = \{X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X)\}$ *xác định trên* U. *Khi đó:* $G\left\langle \left(U, R^\beta\right), \cup, \cap, \sim, \mathscr{T}_0, \mathscr{T}_1 \right\rangle$ *được gọi là một nhóm abel.*

Ví dụ 4.8. Cho các tôpô $\mathscr{T}_0, \mathscr{T}_1, \mathscr{T}_a, \mathscr{T}_b, \mathscr{T}_c \in G$ tương ứng với các quan hệ:

$$R_o^\beta = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \quad R_1^\beta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_a^\beta = \begin{bmatrix} 1 & 1 & 0.8 & 0 & 0 & 0 \\ 1 & 1 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0.8 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}, \quad R_b^\beta = \begin{bmatrix} 1 & 0 & 0.8 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0.8 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$$R_c^\beta = \begin{bmatrix} 1 & 0 & 0.6 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0.6 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad U = \{u_1, u_2, u_3, u_4, u_5, u_6\}. \text{ Khi đó ta có:}$$

$$\mathscr{T}_0 = \left\{ X \subseteq U | \underline{R_0^\beta}(X) = \overline{R_0^\beta}(X) \right\} = \{\emptyset, U\}$$

$$\mathscr{T}_1 = \left\{ X \subseteq U | \underline{R_1^\beta}(X) = \overline{R_1^\beta}(X) \right\} = P(U) = 2^U$$

$$\mathscr{T}_a = \left\{ X \subseteq U | \underline{R_a^\beta}(X) = \overline{R_a^\beta}(X) \right\} = \{\emptyset, \{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, U\}$$

$$\mathscr{T}_b = \left\{ X \subseteq U | \underline{R_b^\beta}(X) = \overline{R_b^\beta}(X) \right\}$$

$$= \left\{ \begin{array}{l} \emptyset, \{u_1, u_2, u_3\}, \{u_4, u_5, u_6\}, \{u_1, u_3, u_4, u_5\}, \{u_2, u_6\}, \{u_2\}, \\ \{u_1, u_3, u_4, u_5, u_6\}, \{u_6\}, \{u_1, u_2, u_3, u_4, u_5\}, \{u_2, u_6\}, \{u_1, u_3, u_4, u_5\}, U \end{array} \right\}$$

(1): Vì $\left(R_a^\beta \cap R_b^\beta\right) \cap R_c^\beta = R_a^\beta \cap \left(R_b^\beta \cap R_c^\beta\right) = R_c^\beta$, do đó $(\mathscr{T}_a \cup \mathscr{T}_b) \cup \mathscr{T}_c = \mathscr{T}_a \cup (\mathscr{T}_b \cup \mathscr{T}_c) = \mathscr{T}_c$;

(2): Vì $\left(R_a^\beta \cup R_b^\beta\right) \cup R_c^\beta = R_a^\beta \cup \left(R_b^\beta \cup R_c^\beta\right) = R_a^\beta$, do đó $(\mathscr{T}_a \cup \mathscr{T}_b) \cup \mathscr{T}_c = \mathscr{T}_a \cup (\mathscr{T}_b \cup \mathscr{T}_c) = \mathscr{T}_a$;

(3): Vì $\left(R_a^\beta \cup R_b^\beta\right) \cap R_c^\beta = R_a^\beta \cap R_c^\beta \cup R_b^\beta \cap R_c^\beta = R_c^\beta$, do đó $(\mathscr{T}_a \cup \mathscr{T}_b) \cup \mathscr{T}_c = \mathscr{T}_a \cup (\mathscr{T}_b \cup \mathscr{T}_c) = \mathscr{T}_c$;

(4): Vì $R_a^\beta \cap R_b^\beta = R_b^\beta \cap R_a^\beta = R_b^\beta$, do đó $\mathscr{T}_a \cup \mathscr{T}_b = \mathscr{T}_b \cup \mathscr{T}_a = \mathscr{T}_b$;

(5): Vì $R_a^\beta \cup R_b^\beta = R_b^\beta \cup R_a^\beta = R_a^\beta$, do đó $\mathscr{T}_a \cap \mathscr{T}_b = \mathscr{T}_b \cap \mathscr{T}_a = \mathscr{T}_a$;

(6): Vì $R_0^\beta \cap R_a^\beta = R_a^\beta$, do đó $\mathscr{T}_0 \cup \mathscr{T}_a = \mathscr{T}_a$; (7): Vì $R_1^\beta \cup R_a^\beta = R_a$, do đó $\mathscr{T}_1 \cap \mathscr{T}_a = \mathscr{T}_a$;

4.3. Đề xuất cấu trúc tôpô Hausdorff

Bảng quyết định số $DT = (U, C, D, f)$ như được trình bày trong Bảng 1.3 có thuộc tính quyết định D là các giá trị rời rạc, tức là các đối tượng giống nhau là không phân biệt được. Khi đó lớp tương đương của các đối tượng này không giao nhau, rõ ràng tôpô trên thuộc tính D là một cấu trúc tôpô Hausdorff. Do đó, để xác định một thuộc tính điều kiện $c \in C$ có ảnh hưởng tới thuộc tính D thì thuộc tính đó cũng phải có cấu trúc tôpô Hausdorff không rỗng. Khi đó, nếu tôpô của một thuộc tính không rỗng sẽ luôn tồn tại một tập mở trong tôpô giao với mỗi phân lớp của D khác rỗng. Theo định nghĩa về độ đo miền dương của RS, các thuộc tính này có liên quan đến tập rút gọn. Do đó, phần này đề xuất xây dựng cấu trúc tôpô Hausdorff, làm cơ sở để chọn lọc các thuộc tính hiệu quả cho tập rút gọn.

Định nghĩa 4.11 (Tính khả li của quan hệ mờ ngưỡng β)**.** Cho không gian xấp xỉ (U, R^β) trong đó R^β là quan hệ tương đương mờ β. Khi đó R^β được gọi là phân biệt được nếu với mọi $p \in U$ tồn tại $q \neq p \in U$ sao cho $[p]_{R^\beta} \cap [q]_{R^\beta} = \emptyset$. Kí hiệu quan hệ này là R_H^β .

Mệnh đề 4.9 (Tôpô Hausdorff từ quan hệ R_H^β)**.** *Cho tôpô* $\mathscr{T}_H = \{X \subseteq U | \underline{R^\beta}(X) = \overline{R^\beta}(X)\}$ *xác định trên* U*. Khi đó,* $\mathscr{T}_H$ *được gọi là tôpô Hausdorff nếu* R^β *là một* R_H^β *.*

Chứng minh. Ta cần chứng minh hai điều kiện sau đây thỏa mãn:

(1): Chứng minh tương tự như mệnh đề 4.1, ta có $\mathscr{T}_H$ là một tôpô xác định trên U;

(2): Giả sử $X, Y \in \mathscr{T}$, khi đó:

$\underline{R_H^\beta}(X) = \overline{R_H^\beta}(X) \Leftrightarrow \cup_{u\in U}\left\{[u]_{R_H^\beta} | [u]_{R_H^\beta} \subseteq X\right\} = \cup_{u\in U}\left\{[u]_{R_H^\beta} | [u]_{R_H^\beta} \cap X \neq \emptyset\right\}$, và

$\underline{R_H^\beta}(Y) = \overline{R_H^\beta}(Y) \Leftrightarrow \cup_{y\in U}\left\{[y]_{R_H^\beta} | [y]_{R_H^\beta} \subseteq Y\right\} = \cup_{y\in U}\left\{[y]_{R_H^\beta} | [y]_{R_H^\beta} \cap Y \neq \emptyset\right\}$.

Hơn nữa:
$$\begin{aligned} X \cap Y &= \cup_{p,q\in U}\left\{[u]_{R_H^\beta} \cap [y]_{R_H^\beta} | [u]_{R_H^\beta} \subseteq X, [y]_{R_H^\beta} \subseteq X\right\} \\ &= \cup_{p,q\in U}\left\{[u]_{R_H^\beta} \cap [y]_{R_H^\beta} | [u]_{R_H^\beta} \cap X \neq \emptyset, [u]_{R_H^\beta} \cap X \neq \emptyset\right\} \end{aligned}.$$

Khi đó, nếu $[u]_{R_H^\beta} \cap [y]_{R_H^\beta} = \emptyset$ thì $X \cap Y = \emptyset$.

Từ (1) và (2), ta có $\mathscr{T}_H$ là một tôpô Hausdorff. □

Mệnh đề 4.10 (Xác định thuộc tính có quan hệ R_H^β)**.** *Cho bảng quyết định $DT = (U, C, D, f)$ và $c \in C$. Khi đó c được gọi là thuộc tính có quan hệ R_H^β nếu $max_1(V_c) - max_2(V_c) > \beta$. Trong đó V_c là tập giá trị của thuộc tính c.*

Chứng minh. Đặt, $m_1 = max_1(V_c)$ và $m_2 = max_1(V_c)$. Rõ ràng nếu $m_1 - m_2 > \beta$ thì với mọi $m < m_2$ ta luôn có $m_1 - m > \beta$. Hơn nữa, theo công thức 4.1 nếu $m_2 < m_1$ thì $[m_1]_{R^\beta} \cap [m_2]_{R^\beta} = \emptyset$, do đó $[m_1]_{R^\beta} \cap [m]_{R^\beta} = \emptyset$. ta có đpcm. □

4.4. Rút gọn thuộc tính trong bảng quyết định theo tiếp cận tôpô Hausdorff

4.4.1. Đề xuất thuật toán tìm tập rút gọn trong bảng quyết định theo phương pháp lai ghép filter - wrapper, sử dụng cấu trúc tôpô Hausdorff

Định nghĩa 4.12 (Thuộc tính quan trọng theo tiếp cận tôpô Hausdorff)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và $c \in C$. Khi đó c được gọi là thuộc tính quan trọng với D nếu $\mathscr{T}_c$ là một tôpô Hausdorff.

Định nghĩa 4.13 (Đồng cấu trúc phụ thuộc)**.** Cho bảng quyết định $DT = (U, C, D, f)$ và hai tôpô $\mathscr{T}_p$, $\mathscr{T}_q$ xác định trên U tương ứng với $p, q \in C$. Khi đó $\mathscr{T}_p$ được gọi là

đồng cấu trúc phụ thuộc với $\mathscr{T}_q$ nếu $\mathscr{T}_p \cup \mathscr{T}_D = \mathscr{T}_q \cup \mathscr{T}_D$.

Sau đây là thuật toán rút gọn thuộc tính theo phương pháp phân cụm và đánh giá các nhóm thuộc tính Hausdorff.

Thuật toán 4.1 Thuật toán rút gọn thuộc tính theo tiếp cận filter - wrapper các cụm thuộc tính (CFW).

Input Bảng quyết định $DT = (U, C, D)$ với $\Delta = \{0.1, 0.2, ..., 0.8, 0.9\}$ và mô hình phân lớp $Model$

Output Tập rút gọn R

1: $R = \emptyset$;
2: **for all** $\beta \in \Delta$ **do**
3: $H^\beta \leftarrow \emptyset$;
4: $CH^\beta \leftarrow \emptyset$;
5: $R^\beta \leftarrow \emptyset$;
6: **for all** $c \in C$ **do**
7: **if** $max_1(V_c) - max_2(V_c) > \beta$ **then**
8: $H^\beta = H^\beta \cup \{c\}$; {Filter các thuộc tính Hausdorff}
9: **end if**
10: **end for**
11: **for all** $p \in \{H^\beta - CH^\beta\}$ **do**
12: $U_p = \emptyset$;
13: **for all** $q \in \{H^\beta - CH^\beta - p\}$ **do**
14: **if** $\mathscr{T}_p \cup \mathscr{T}_D = \mathscr{T}_q \cup \mathscr{T}_D$ **then**
15: $U_p = U_p \cup \{q\}$; {Phân cụm thuộc tính Hausdorff}
16: **end if**
17: **end for**
18: $CH^\beta = CH^\beta \cup U_p$;
19: **if** $ACC_{U_p}^{Model} > ACC_{R^\beta}^{Model}$ **then**
20: $R^\beta = U_p$; {Wrapper các cụm thuộc tính Hausdorff}
21: **end if**
22: **end for**
23: **if** $ACC_{R^\beta}^{Model} > ACC_{R}^{Model}$ **then**
24: $R = R^\beta$; {Wrapper các tập rút gọn ứng viên β}
25: **end if**
26: **end for**
27: **return** R;

Trong thuật toán này, kí hiệu H^β là tập các thuộc tính Hausdorff thu được từ tập thuộc tính C ban đầu trong bảng quyết định DT. CH^β là các cụm thuộc tính được

phân loại từ H^β, trong đó mỗi cụm thuộc tính U_p là các thuộc tính $q \in H^\beta$ có cùng cấu trúc phụ thuộc với thuộc tính p theo định nghĩa 4.13.

Tiếp theo sẽ là phần đánh giá độ phức tạp của thuật toán CFW. Kí hiệu $|U|$ là số các đối tượng, $|C|$ là số các thuộc tính, $|H^\beta|$ là số các thuộc tính Hausdorff, và $|CH^\beta|$ là số các nhóm thuộc tính Hausdorff có cùng cấu trúc phụ thuộc. Khi đó độ phức tạp từ 6-10 là $\mathcal{O}(2|U||C|)$, độ phức tạp từ 11-22 là $\mathcal{O}(|U|^2|H^\beta|^2)$. Giả sử $\mathbb{T}$ là thời gian thực hiện của mô hình phân lớp $Model$. Với số lượng Δ rất nhỏ, khi đó độ phức tạp của thuật toán là $\mathcal{O}(2|U||C|) + \mathcal{O}(|U|^2|H^\beta| + |CH^\beta|\mathbb{T})$.

Ví dụ 4.9. Xét bảng quyết định $DT = (U, C, D, f)$ như được trình bày trong Bảng 1.3. Xét $\beta = 0.7$ ta có:

Bước 1: $H^{0.7} \leftarrow \emptyset$; $CH^\beta \leftarrow \emptyset$; $R^\beta \leftarrow \emptyset$;

Bước 2: Xác định các thuộc tính Hausdorff dựa trên mệnh đề 4.10 ta có $H^{0.7} = \{a, c, d, e, f\}$;

Bước 3: Tính các ma trận quan hệ của các thuộc tính trong $H^{0.7}$ theo công thức quan hệ 4.1 với $\beta = 0.7$ ta có:

$$R_a^{0.7} = \begin{bmatrix} 1 & 1 & 0.8 & 0 & 0 & 0 \\ 1 & 1 & 0.8 & 0 & 0 & 0 \\ 0.8 & 0.8 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix} \quad R_c^{0.7} = \begin{bmatrix} 1 & 0 & 0.8 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0.8 & 0 & 1 & 0.8 & 0.8 & 0 \\ 1 & 0 & 0.8 & 1 & 1 & 0 \\ 1 & 0 & 0.8 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_d^{0.7} = \begin{bmatrix} 1 & 0.8 & 0.8 & 1 & 1 & 0 \\ 0.8 & 1 & 0 & 0.8 & 0.8 & 0 \\ 0.8 & 0 & 1 & 0.8 & 0.8 & 0 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad R_D = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$R_f^{0.7} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0.8 & 0.8 & 0.8 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \end{bmatrix} \quad R_e^{0.7} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0.8 & 0.8 & 0.8 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \\ 0 & 0.8 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Vì $R_a^{0.7} \cap R_D^{0.7} = R_c^{0.7} \cap R_D^{0.7} \Leftrightarrow \mathscr{T}_a \cup \mathscr{T}_D = \mathscr{T}_c \cup \mathscr{T}_D$, do đó $CH^{0.7} = \emptyset \cup \{a,c\} = \{\{a,c\}\}$; $R_e^{0.7} \cap R_D^{0.7} = R_f^{0.7} \cap R_D^{0.7} \Leftrightarrow \mathscr{T}_e \cup \mathscr{T}_D = \mathscr{T}_f \cup \mathscr{T}_D$.

Do đó $CH^{0.7} = \{a,c\} \cup \{e,f\} = \{\{a,c\},\{e,f\}\}$;

Cuối cùng thuộc tính d được bổ xung vào $CH^{0.7}$, do đó $CH^{0.7} = \{\{a,c\},\{e,f\},\{d\}\}$.

Bước 4: wrapper từng nhóm thuộc tính trên mô hình phân lớp $Model$, nhóm thuộc tính nào có độ chính xác phân lớp cao nhất thì nhóm đó được gán cho $R^{0.7}$

Bước 5: Giả sử $R^{0.7}$ có độ chính xác phân lớp cao nhất trong các ngưỡng δ. Khi đó tập rút gọn thu được của thuật toán $R = R^{0.7}$.

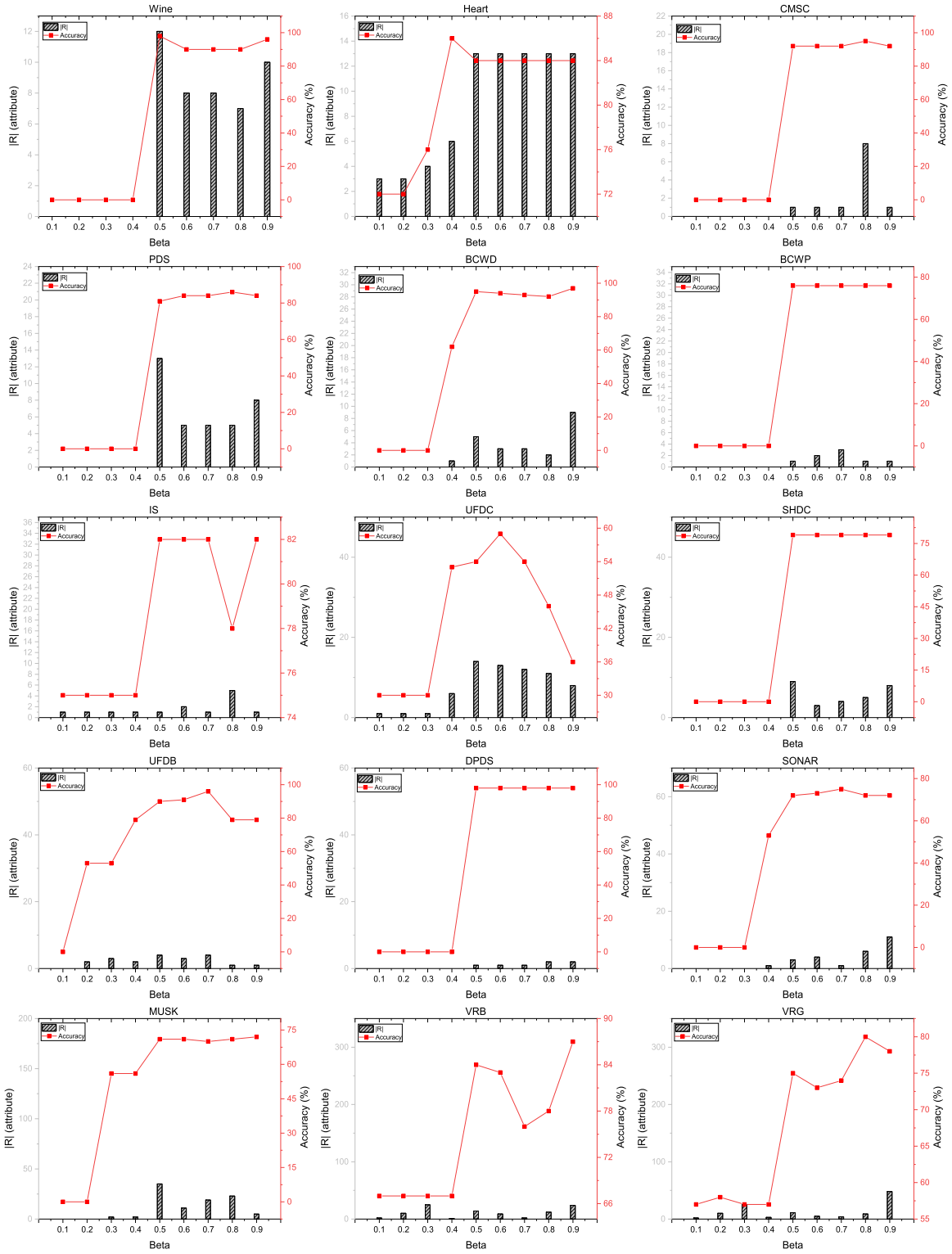
4.4.2. Thực nghiệm và đánh giá thuật toán

Mục tiêu của phần thực nghiệm nhằm đánh giá tính hiệu quả của thuật toán đề xuất khi áp dụng với các bộ dữ liệu trong thực tiễn. Trên cơ sở đó có thể khẳng định tính đúng đắn của khung nền tảng lý thuyết bài toán rút gọn thuộc tính theo tiếp cận tôpô Hausdorff. Sau đây là kế hoạch thực nghiệm thuật toán đề xuất.

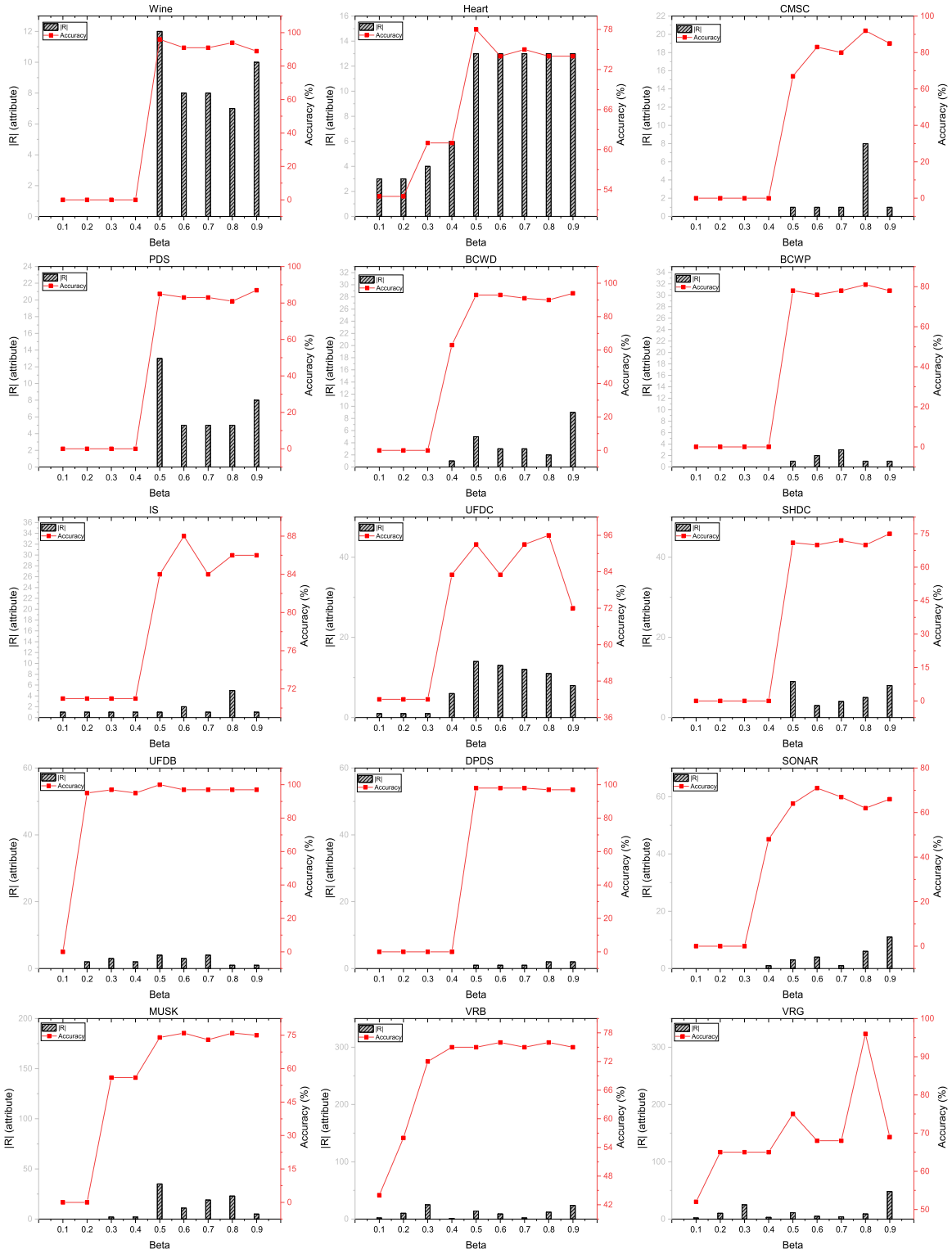
4.4.2.1. Kịch bản và môi trường thực nghiệm

1) Chọn lựa giá trị β. Mục tiêu của quá trình thực nghiệm này nhằm tìm kiếm giá trị β phù hợp nhất cho từng bộ dữ liệu của thuật toán đề xuất, trong đó giá trị β được chọn trong khoảng $[0.1, 0.9]$ với mỗi bước nhảy là 0.1.

2) Đánh giá thuật toán đề xuất. Sau khi chọn lựa được các giá trị β phù hợp, thực hiện so sánh và đánh giá thuật toán đề xuất với các thuật toán rút gọn thuộc tính điển



Hình 4.1: *Biểu đồ phân tích mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn tại mỗi giá trị β trên mô hình phân lớp SVM.*



Hình 4.2: *Biểu đồ phân tích mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn tại mỗi giá trị β trên mô hình phân lớp k-NN.*

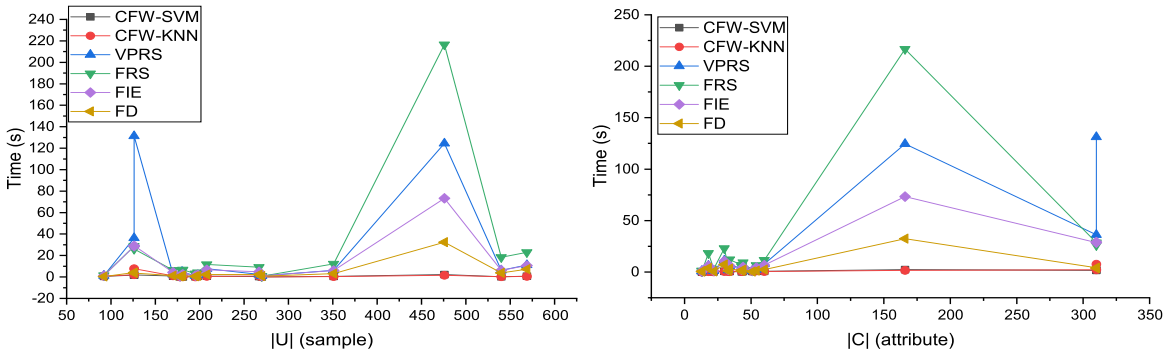
Bảng 4.1: *Mô tả các tập dữ liệu thực nghiệm*

STT	Tập dữ liệu	Mô tả	$\|U\|$	$\|C\|$	$\|D\|$
1	Wine	Wine	178	13	3
2	Heart	Statlog (Heart)	270	13	2
3	CMSC	Climate Model Simulation Crashes Data Set	540	18	2
4	PDS	Parkinsons Data Set	196	22	2
5	BCWD	Breast Cancer Wisconsin (Diagnostic)	569	30	2
6	BCWP	Breast Cancer Wisconsin (Prognostic)	198	33	2
7	IS	Ionosphere	351	34	2
8	UFDC	Ultrasonic flowmeter diagnostics (C)	181	43	4
9	UFDD	Ultrasonic flowmeter diagnostics (D)	181	43	4
10	SHDC	SPECTF Heart Data Set	267	44	2
11	UFDB	Ultrasonic flowmeter diagnostics (B)	92	51	3
12	DPDS	Divorce Predictors data set	170	54	2
13	Sona	Connectionist Bench	208	60	2
14	Musk	Musk	476	166	2
15	VRB	Voice Rehabilitation(Binary)	126	310	2
16	VRG	Voice Rehabilitation(Gender)	126	310	2

hình trên tiếp cận độ đo gồm có: (1) thuật toán rút gọn thuộc tính theo tiếp cận tập thô với độ chính xác điều chỉnh (VPRS) [131]; (2) thuật toán rút gọn thuộc tính theo tiếp cận tập thô mờ (FRS) [68]; (3) thuật toán rút gọn thuộc tính theo tiếp cận Entropy thông tin mờ (IFE) [99]; (4) thuật toán rút gọn thuộc tính theo tiếp cận khoảng cách mờ (FD) [33].

Tất cả các thuật toán đều được đánh giá trên 16 bộ dữ liệu được tải về từ kho dữ liệu học máy UCI. Các tập dữ liệu được chọn là các tập dữ liệu có thuộc tính quyết định miền giá trị số và thuộc tính điều kiện có miền giá trị rời rạc. Các thuật toán được cài đặt bằng ngôn ngữ lập trình Python trên nền hệ điều hành Window 10 với cấu hình phần cứng là bộ xử lý Core-i5, bộ nhớ RAM 8G.

Các tập dữ liệu được sắp xếp theo trật tự tăng dần về số lượng thuộc tính điều kiện. Chi tiết các tập dữ liệu được mô tả trong Bảng 4.1 trong đó kí hiệu $|U|$ là số lượng mẫu, kí hiệu $|C|$ là số lượng các thuộc tính điều kiện và $|D|$ là số phân lớp trong thuộc tính quyết định. Trong các bộ dữ liệu thực nghiệm, bộ dữ liệu UFDC và Sonar là các



Hình 4.3: *Biểu đồ phân tích mối quan hệ giữa thời gian thực hiện của thuật toán và $|U|$ (hình trái), giữa thời gian thực hiện của thuật toán và $|C|$ (hình phải).*

bộ dữ liệu nhiễu, có độ chính xác phân lớp ban đầu trên mô hình huấn luyện *Model* thấp $< 70\%$. Các tiêu chí đánh giá gồm: thời gian thực hiện của thuật toán (seconds), kích thước của tập rút gọn ($|R|$), và độ chính xác phân lớp của tập dữ liệu trên mô hình phân lớp dữ liệu *Model* (percentage). Quá trình thực nghiệm cũng coi trọng khả năng loại bỏ nhiễu của các thuật toán trên các tập dữ liệu xấu. Trước khi thực hiện thuật toán rút gọn thuộc tính, các tập dữ liệu được chuẩn hóa giá trị về đoạn $[0, 1]$ để nâng cao hiệu năng cho thuật toán và các mô hình phân lớp.

Mỗi thuật toán được thực hiện 10 lần trên từng bộ dữ liệu với 90% dữ liệu được lấy ngẫu nhiên từ tập dữ liệu gốc. Hai mô hình phân lớp được sử dụng để đánh giá gồm có mô hình phân lớp máy vector hỗ trợ [1] (Support Vector Machine - SVM) và mô hình phân lớp k-láng giềng [2] (k-Nearest Neighbor - kNN, k=|D|). Chỉ số đánh giá độ chính xác (accuracy) và phương pháp đánh giá chéo 10-folde được kết hợp để đánh giá chất lượng phân lớp của tập rút gọn.

4.4.2.2. Chọn lọc giá trị β cho mỗi tập dữ liệu

Trước khi thực hiện so sánh thuật toán đề xuất với các thuật toán rút gọn thuộc tính khác, ta cần lựa chọn giá trị β phù hợp nhất trên từng bộ dữ liệu khác nhau cho thuật toán. Với mỗi tập dữ liệu thực nghiệm, thực nghiệm thuật toán với từng giá trị β khác

[1] https://brilliant.org/wiki/support-vector-machines/
[2] https://brilliant.org/wiki/k-nearest-neighbors/

Bảng 4.2: *So sánh kích thước của tập rút gọn thu được từ các thuật toán*

STT	Tập dữ liệu	|R|						
		$\|C\|$	CFW-SVM	CFW-kNN	VPRS	FRS	FIE	FD
1	wine	13	10.8	7.6	11.8	10.4	10.6	7.1
2	heart	13	6.7	5.5	11.5	13.9	10.2	6.7
3	CMSC	20	8.2	8.7	9.5	20.3	20.1	3.5
4	PDS	22	5.2	4.4	9.4	8.5	10.8	4.3
5	BCWD	30	3.2	3.6	14.8	7.6	12.1	4.1
6	BCWP	32	2.9	2.2	8.9	12.6	12.4	5.8
7	IS	34	2.1	2.1	20.9	11.3	19.6	6.1
8	UFDC	43	13.9	4.3	15.3	8.7	11.7	5.2
9	UFDD	43	5.1	3.6	19.9	6.6	8.3	3.3
10	SHDC	44	3.1	2.2	44.3	10.3	14.7	5.9
11	UFDB	51	4.1	3.4	8.9	5.8	11.9	5.2
12	DPDS	54	2.5	1.6	8.4	15.7	24.4	4.4
13	sonar	60	4.6	7.4	44.3	17.5	25.2	7.6
14	musk	166	5.7	11.4	86.6	23.9	29.5	8.8
15	VRB	310	9.1	4.3	56.6	18.9	35.8	7.5
16	VRG	310	9.6	2.1	72.4	16.5	36.4	10.6

nhau trong khoảng $[0.1, 0.9]$ với mỗi bước nhảy là 0.1. Như vậy, mỗi bộ dữ liệu được thực nghiệm với 09 lần khác nhau. Với mỗi giá trị β khác nhau, kích thước tập rút gọn sẽ khác nhau và độ chính xác phân lớp với mỗi tập rút gọn cũng có thể khác nhau. Tuy nhiên, với các giá trị β quá nhỏ, có thể sẽ không tồn tại tập rút gọn.

Quan sát biểu đồ của các tập dữ liệu (Wine, CSMC, PDSB, BCWP, ..) trong Hình 4.1 và Hình 4.2 để biết thêm thông tin chi tiết. Hơn nữa, biểu đồ trong các hình này cũng phân tích mối quan hệ giữa kích thước của tập rút gọn và độ chính xác phân lớp trên từng tập dữ liệu. Chúng ta có thể thấy, nhiều giá trị β cho tập rút gọn kích thước nhỏ nhưng độ chính xác lại cao hơn so với các giá trị δ khác. Tuy nhiên môi quan hệ nay không tuyến tính nên chúng ta phải cân nhắc giá trị β cho phù hợp với mục tiêu về độ chính xác hay kích thước của tập rút gọn. Ở đây, lựa chọn giá trị β sao cho đảm bảo tính cân bằng giữa độ chính xác phân lớp và kích thước của tập rút gọn thu được từ thuật toán đề xuất.

Bảng 4.3: *So sánh độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán trên mô hình phân lớp SVM*

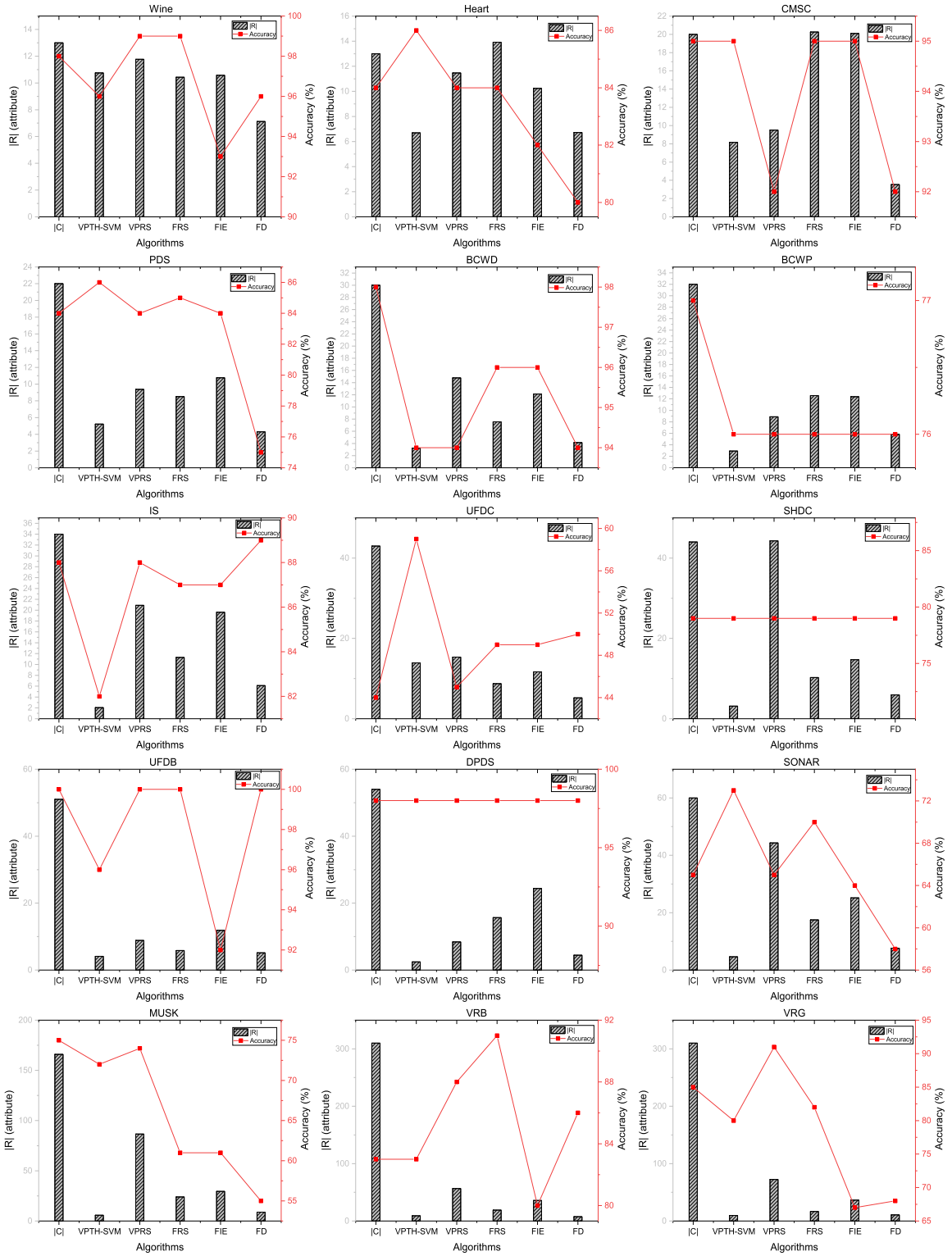
STT	Tập dữ liệu	Độ chính xác phân lớp (%)					
		RAW	CFW-SVM	VPRS	FRS	FIE	FD
1	wine	98±0.7	96±0.9	99±0.6	99±0.3	93±0.1	96±0.8
2	heart	84±0.8	86±0.6	84±0.3	84±0.3	82±0.9	80±0.7
3	CMSC	95±0.8	95±0.4	92±0.4	95±0.1	95±0.8	92±0.6
4	PDS	84±0.7	86±0.6	84±0.7	85±0.9	84±0.7	75±0.8
5	BCWD	98±0.6	94±0.7	94±0.2	96±0	96±0.8	94±0.7
6	BCWP	77±0.3	76±0.3	76±0.6	76±0.2	76±0.8	76±0
7	IS	88±0.5	82±1	88±0.9	87±0.5	87±0.3	89±0.6
8	UFDC	44±0.8	**59±0.7**	45±0.5	49±0.1	49±0.6	50±1
9	UFDD	68±0.8	63±0.5	68±0.1	64±1	63±0.7	62±0.5
10	SHDC	79±0.5	79±1	79±0	79±0	79±0.6	79±0.3
11	UFDB	99±0.4	96±0.9	99±0.6	99±0.2	92±0.8	99±0.2
12	DPDS	98±0.6	98±0.3	98±0.3	98±0.6	98±0.4	98±0.5
13	sonar	65±0.8	**73±0.2**	65±0.2	70±0.7	64±0	58±0
14	musk	75±0.3	72±0.2	74±0.8	61±0.4	61±0.1	55±0.4
15	VRB	83±0.1	83±0.2	88±0.6	91±0.4	80±0.8	86±1
16	VRG	85±0.9	80±0.2	91±0.7	82±0.5	67±0.2	68±0.4

4.4.2.3. Đánh giá thuật toán đề xuất

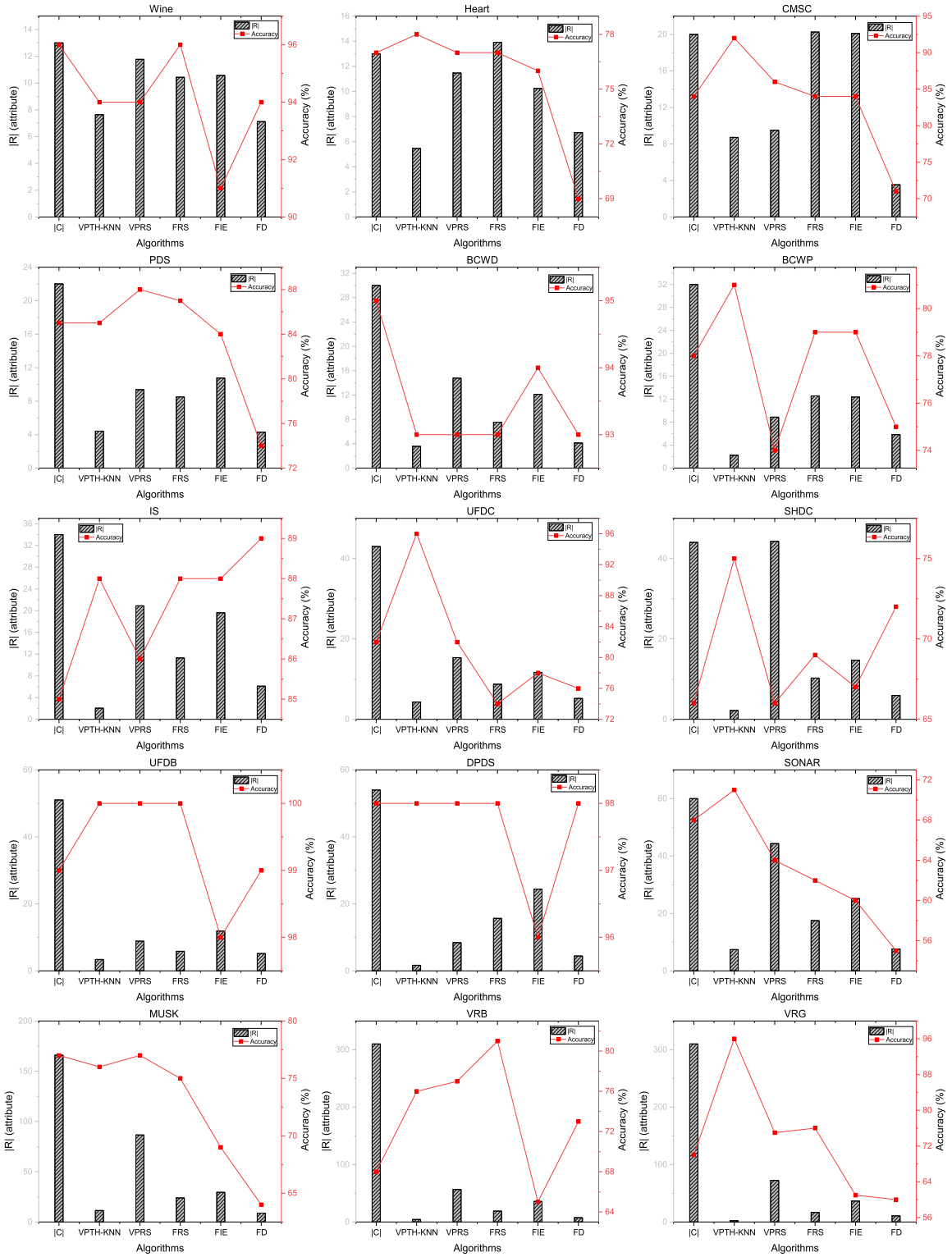
Giai đoạn wrapper của thuật toán đề xuất sử dụng hai mô hình phân lớp là SVM và k-NN(k=|D|). Trước tiên, chương này đánh giá hiệu năng của thuật toán đề xuất trên mô hình phân lớp SVM.

Đánh giá thuật toán trên mô hình phân lớp SVM

Các kết quả thực nghiệm của thuật toán trên mô hình phân lớp SVM được phân tích và thống kê về kích thước tập rút gọn thu được trình bày trong Bảng 4.2, về độ chính xác phân lớp được trình bày trong Bảng 4.3, và thời gian thực hiện được trình bày trong Bảng 4.5. Quan sát dữ liệu trong Bảng 4.2 ta có thể thấy thuật toán đề xuất CFW và thuật toán rút gọn thuộc tính theo tiếp cận khoảng cách mờ FD cho tập rút gọn có kích thước tốt nhất, tuy nhiên độ chính xác phân lớp trên tập rút gọn của thuật toán đề xuất hoàn toàn tốt hơn thuật toán theo tiếp cận FD. Bảng 4.3 cho thấy thuật



Hình 4.4: *Biểu đồ phân tích mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn của mỗi thuật toán trên mô hình phân lớp SVM.*



Hình 4.5: *Biểu đồ phân tích mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn của mỗi thuật toán trên mô hình phân lớp KNN.*

Bảng 4.4: *So sánh độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán trên mô hình phân lớp KNN*

STT	Tập dữ liệu	Độ chính xác phân lớp (%)					
		RAW	CFW-kNN	VPRS	FRS	FIE	FD
1	wine	96±0.2	94±0.1	94±0.1	96±0.9	91±0.4	94±0.6
2	heart	77±0.5	78±0.1	77±0.3	77±0.3	76±0.2	69±0.7
3	CMSC	84±0.1	92±0.1	86±0.2	84±0.6	84±0.9	71±0.1
4	PDS	85±0.7	85±0.3	88±0.9	87±0.1	84±0.3	74±0.5
5	BCWD	95±0.2	93±0.1	93±0.9	93±0.9	94±0.7	93±0.7
6	BCWP	78±0.8	81±0.9	74±0.6	79±0.6	79±0.6	75±0.6
7	IS	85±0.6	88±0.6	86±0.9	88±0.7	88±0.4	89±0.4
8	UFDC	82±0.1	96±0.2	82±0.1	74±0.9	78±0.1	76±0.2
9	UFDD	81±0.5	81±0.9	77±0.9	77±0.5	82±0.6	72±0.7
10	SHDC	66±0.1	75±0.7	66±0.5	69±0.8	67±1	72±0.6
11	UFDB	99±0.8	100	100	100	98±0	99±0.5
12	DPDS	98±0.4	98±0	98±0.3	98±0.4	96±0.9	98±0.2
13	sonar	68±0.3	71±0.3	64±0.5	62±0.7	60±0.7	55±0.3
14	musk	77±0.5	76±0.7	77±0.1	75±1	69±0.4	64±0.3
15	VRB	68±0.3	76±0.6	77±0.1	81±0.3	65±0.1	73±0.1
16	VRG	70±0.8	96±0.4	75±0.8	76±0.9	61±1	60±0.9

toán đề xuất CFW và hai thuật toán dựa trên độ đo miền dương là VPRS và FRS cho tập rút gọn có độ chính xác tốt nhất, hầu như không chênh lệch so với tập dữ liệu gốc. Tuy nhiên kích thước tập rút gọn thu được từ thuật toán đề xuất hoàn toàn vượt trội so với hai thuật toán theo tiếp cận VPRS và FRS. Đặc biệt là các bộ dữ liệu nhiễu (**UFDS**, **Sonar**), độ chính xác phân lớp được cải thiện từ 44% to 59%. Bảng 4.5 cho thấy thuật toán đề xuất có thời gian hoàn toàn vượt trội so với các thuật toán khác.

Quan sát biểu đồ trong Hình 4.4 ta có thể thấy mối quan hệ về kích thước và độ chính xác phân lớp của các tập rút gọn thu được từ các thuật toán, hầu hết các tập rút gọn của thuật toán đề xuất đều có kích thước nhỏ hơn nhưng độ chính xác phân lớp không chênh lệch so với các thuật toán tốt nhất. Quan sát biểu đồ của các tập dữ liệu (**Heart**, **CMCS**, **PDS**, **BCWP**, **UFDC**, **SHDC**, **DPDS**, **Sonar**) ta có thể thấy tính hiệu quả về kích thước và độ chính xác phân lớp của thuật toán đề xuất là hoàn toàn vượt trội so với các thuật toán khác. Qua đó ta có thể thấy nhiều tập rút gọn có kích

Bảng 4.5: *So sánh thời gian thực hiện của các thuật toán*

STT	Tập dữ liệu	Thời gian thực hiện (s)					
		CFW-SVM	CFW-kNN	VPRS	FRS	FIE	FD
1	wine	0.05	0.07	0.7	1.27	0.58	0.3
2	heart	0.08	0.1	1.08	0.79	1.03	0.63
3	CMSC	0.28	0.26	6.19	18.28	6.14	3.79
4	PDS	0.17	0.09	1.24	3.09	1.23	0.6
5	BCWD	0.67	0.49	11.19	23.01	10.69	7.3
6	BCWP	0.71	0.57	1.81	4.13	2.14	1.04
7	IS	0.6	0.43	6.13	12.17	6.08	3.17
8	UFDC	0.11	0.49	4.23	6.73	2.49	1.17
9	UFDD	0.18	0.62	4.84	5.43	2.11	1.12
10	SHDC	0.45	0.85	2.41	9.16	4.8	2.29
11	UFDB	0.56	0.77	1.22	1.39	1.22	0.38
12	DPDS	0.82	1.06	2.41	6.42	4.34	1.31
13	sonar	0.63	0.52	7.8	11.63	6.65	2.25
14	musk	2.26	1.55	124.48	216.6	73.32	32.31
15	VRB	1.7	2.12	36.25	28.67	28.59	4.1
16	VRG	1.93	7.67	131.24	26.09	29.11	3.63

thước lớn nhưng chưa chắc đã có độ chính xác cao hơn.

Tóm lại, thuật toán đề xuất thực hiện trên mô hình phân lớp SVM cho tập rút gọn có kích thước và độ chính xác phân lớp là không chênh lệch đáng kể so với các thuật toán tốt nhất. Tuy nhiên thời gian thực hiện của thuật toán đề xuất là hoàn toàn vượt trội so với các thuật toán khác.

Đánh giá thuật toán trên mô hình phân lớp KNN

Kết quả thực nghiệm trên mô hình phân loại kNN của thuật toán đề xuất được mô tả chi tiết trong Bảng 4.2, 4.4 và 4.5. Kết quả trung bình của các tiêu chí đánh giá về kích thước, độ chính xác phân lớp và thời gian thực hiện của thuật toán đề xuất đều có kết quả vượt trội so với các thuật toán khác. Quan sát Bảng 4.2 ta có thể thấy kích thước trung bình của tập rút gọn thu được từ thuật toán đề xuất có kết quả tốt nhất và tốt hơn khi thực hiện trên mô hình phân lớp SVM. Quan sát Bảng 4.4 và Bảng4.5 ta thấy độ chính xác phân lớp của tập rút gọn và thời gian thực hiện của thuật toán đề xuất là hoàn toàn vượt trội so với các thuật toán tốt nhất.

Quan sát biểu đồ trong Hình 4.4 ta có thể thấy mối quan hệ về kích thước và độ chính xác phân lớp của các tập rút gọn thu được từ các thuật toán, hầu hết các tập rút gọn của thuật toán đề xuất đều có kích thước nhỏ hơn nhưng độ chính xác phân lớp lại cao hơn so với các thuật toán tốt nhất. Quan sát biểu đồ của các tập dữ liệu (**Heart**, **CMCS**, **BCWP**, **IS**, **UFDC**, **UFDD** , **SHDC**, **UFDB**, **DPDS**, **Sonar**, **Musk**, **VRG**). Đặc biệt bộ dữ liệu nhiễu (**VRG**) cải thiện độ chính xác phân lớp từ 70% lên 96%.

Bảng 4.6: *Mô tả tập rút gọn thu được từ các thuật toán*

STT	Tập dữ liệu	Tập rút gọn của các thuật toán					
		CFW-SVM	CFWK-NN	VPRS	FRS	FIE	FD
1	wine	[0, 2, 3, 4, 5, 7, 8, 9, 10, 11]	[0, 4, 5, 6, 7, 8, 10]	[12, 11, 0, 9, 6, 1, 3, 10, 7, 5, 2]	[12, 9, 5, 0, 1, 11, 3, 4, 2, 6]	[7, 1, 5, 3, 4, 8, 0, 11, 9, 10]	[11, 12, 7, 0, 1, 5, 9]
2	heart	[1, 2, 5, 8, 11, 12]	[2, 3, 4, 9, 11]	[12, 2, 11, 6, 8, 10, 1, 5, 7, 3, 9]	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]	[6, 1, 8, 12, 5, 10, 2, 11, 0, 3]	[12, 6, 8, 1, 2, 10]
3	CMSC	[2, 3, 4, 5, 8, 9, 14, 15]	[2, 3, 4, 5, 8, 9, 14, 15]	[2, 3, 19, 9, 11, 7, 12, 18, 14]	[2, 3, 15, 0, 5, 1, 6, 4, 7, 9, 13,...]	[0, 1, 4, 6, 5, 7, 10, 8, 9, 3, ...]	[0, 1, 5]

Tiếp theo trang sau

Bảng 4.6 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn của các thuật toán					
		CFW-SVM	CFWK-NN	VPRS	FRS	FIE	FD
4	PDS	[16, 17, 18, 19, 21]	[15, 18, 19, 20]	[0, 2, 18, 17, 19, 20, 21, 9, 11]	[18, 0, 10, 16, 17, 2, 19, 20]	[16, 2, 17, 1, 0, 10, 3, 19, 6, 20]	[0, 16, 2, 1]
5	BCWD	[0, 20, 27]	[0, 20, 27]	[27, 7, 20, 25, 6, 28, 16, 0, 17, ...]	[20, 27, 1, 11, 4, 8, 21]	[9, 21, 27, 6, 8, 11, 3, 15, 1, 18, ..]	[27, 7, 6, 20]
6	BCWP	[0, 26]	[11, 31]	[23, 16, 15, 0, 19, 20, 24, 22]	[0, 4, 2, 5, 6, 19, 11, 9, 12, .. , 20]	[0, 31, 6, 19, 1, 11, 22, 5, 20, 8, 12, 18]	[0, 1, 31, 23, 19]
7	IS	[4, 26]	[4, 26]	[0, 4, 2, 7, 9, 5, 27, 23, 29, 13, ...]	[0, 4, 2, 5, 27, 30, 7, 3, 9, 16, 17]	[14, 0, 27, 28, 7, 31, 18, 23, 26, 4, ...]	[14, 0, 4, 24, 22, 27]
8	UFDC	[0, 8, 9, 10, 25, 27, 28, 29, 30, 31, 32, 33, 34]	[29, 30, 31, 32]	[23, 27, 5, 25, 36, 32, 12, 11, 35, 21, ...]	[7, 9, 25, 5, 27, 0, 39, 11]	[3, 23, 8, 25, 0, 2, 27, 15, 29, 33, 31]	[3, 23, 27, 7, 25]

Tiếp theo trang sau

Bảng 4.6 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn của các thuật toán					
		CFW-SVM	CFWK-NN	VPRS	FRS	FIE	FD
9	UFDD	[10, 27, 28, 41, 42]	[25, 29, 30]	[33, 27, 6, 40, 5, 30, 12, ...]	[25, 27, 17, 3, 11, 42]	[5, 27, 21, 42, 39, 31, 0, 41]	[33, 5, 27]
10	SHDC	[0, 21, 22]	[25, 41]	[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ...]	[40, 29, 1, 2, 13, 25, 3, 4, 18, 9]	[43, 3, 18, 1, 12, 14, 21, 9, ...]	[43, 24, 41, 29, 36]
11	UFDB	[17, 22, 41, 42]	[19, 39, 40]	[41, 14, 4, 32, 15, 29, 34, 23]	[41, 14, 13, 16, 12]	[35, 6, 12, 39, 31, 0, 19, 22, ..]	[41, 14, 16, 20, 7]
12	DPDS	[10, 19]	[18]	[16, 39, 25, 43, 8, 24, 40, 27]	[10, 32, 30, 0, 48, 15, 6, 39, 35, ...]	[44, 45, 3, 51, 34, 5, 30, 46, 41, ...]	[39, 34, 10, 43]
13	sonar	[11, 12, 29, 32]	[9, 16, 17, 43, 47, 52, 53]	[35, 20, 11, 19, 44, 7, 36, 16, ...]	[0, 11, 15, 36, 26, 19, 21, 9, ...]	[19, 25, 16, 22, 34, 27, ...]	[19, 35, 16, 22, 25, 34, 28]

Tiếp theo trang sau

Bảng 4.6 – Tiếp theo trang trước

STT	Tập dữ liệu	Tập rút gọn của các thuật toán					
		CFW-SVM	CFWK-NN	VPRS	FRS	FIE	FD
14	musk	[62, 101, 161, 163, 164]	[12, 20, 50, 62, 93, 104, 109, 110, 128, 140, 161]	[91, 36, 76, 57, 15, 31, 156, 162, 135, 83, ...]	[49, 16, 147, 3, 96, 1, 38, 13, 31, 87, ...]	[16, 124, 1, 36, 23, 46, 60, 68, 131, 15, ...]	[31, 16, 40, 131, 124, 63, 64, 36]
15	VRB	[58, 83, 92, 107, 117, 121, 152, 197, 279]	[79, 84, 270, 271]	[72, 79, 26, 70, 52, 77, 83, ...]	[54, 84, 79, 83, 91, 41, 3, ...]	[58, 70, 59, 60, 62, 92, ...]	[79, 58, 70, 82, 59, 84, 69]
16	VRG	[75, 92, 117, 124, 196, 197, 198, 279, 309]	[124, 309]	[60, 69, 62, 30, 85, 86, 77, 25, 70, ...]	[90, 95, 31, 102, 59, 73, 16, 82, 89,...]	[59, 16, 70, 62, 57, 92, 69, 127, ...]	[79, 58, 70, 86, 84, 121, 56, 69, 16, 62]

Nhìn chung, các kết quả thực nghiệm của thuật toán đề xuất trên mô hình phân lớp kNN đều có kết quả tốt hơn so với các thuật toán còn lại. Trong đó khả năng cải thiện nhiễu và thời gian thực hiện của thuật toán là hoàn toàn vượt trội. Dưới góc nhin thực nghiệm, ta có thể khẳng định tiếp cận rút gọn thuộc tính theo tiếp cận tôpô là hoàn toàn phù hợp. Tiếp cận này có khả năng tạo ra những thay đổi lớn trong cải thiện hiệu năng cho các thuật toán rút gọn thuộc tính. Sau đây là phần phân tích các nguyên nhân ảnh hưởng tới thời gian thực hiện của thuật toán, độ chính xác phân lớp và kích thước

tập rút gọn thu được từ thuật toán.

1) Thời gian tính toán của thuật toán đề xuất.

Như đã trình bày trong phần đánh giá độ phức tạp, thuật toán đề xuất có thời gian tính toán lý thuyết tốt hơn đáng kể so với các thuật toán sử dụng tiếp cận độ đo hiện nay. Hầu hết các thuật toán rút gọn thuộc tính truyền thống đều có độ phức tạp tính toán là $\mathcal{O}(|U|^2|C|^2)$. Khi $|U|$ lớn dẫn đến không gian xấp xỉ sẽ rất lớn, chiếm dụng nhiều tài nguyên lưu trữ và tài nguyên tính toán của hệ thống. Khi $|C|$ lớn, sẽ có nhiều thuộc tính cần phải đánh giá. Ngược lại, độ phức tạp của thuật toán được đề xuất là $\mathcal{O}(2|U||C|) + \mathcal{O}(|U|^2|H^\beta| + |CH^\beta|\mathbb{T})$ có thời gian thực hiện nhỏ hơn đáng kể. Trong đó giai đoạn filter có thời gian chỉ $\mathcal{O}(2|U||C|)$, giai đoạn này là yếu tố chính làm giảm mạnh thời gian thực hiện của thuật toán. Khi $|H^\beta|$ nhỏ thời gia phân cụm các thuộc tính sẽ nhanh. Trong đó thời gian phân cụm của $|H^\beta|$ thuộc tính là $\mathcal{O}(|U|^2|H^\beta|^2)$. Hơn nữa, nếu số cụm phân cụm được ít thì thời gian xác định cụm thuộc tính có độ chính xác phân lớp sẽ nhanh hơn với số lượng cụm lớn. Do đó, khi $|H^\beta|$ càng nhỏ thì thời gian thực hiện của thuật toán càng nhanh và ngược lại.

2) Độ chính xác phân lớp của thuật toán đề xuất.

Hầu hết các phương pháp rút gọn thuộc tính truyền thống đều sử dụng độ đo để đánh giá độ quan trọng của thuộc tính cũng như đo lường lượng thông tin bảo toàn của tập thuộc tính rút gọn so với tập thuộc tính gốc. Tuy nhiên, cách tiếp cận độ đo chủ yếu đánh giá độ tương tự giữa các tập dựa trên tổng thành phần mà không xem xét đến nội dung bên trong thành phần đó. Trong khi đó, cách tiếp cận dựa trên cấu trúc tôpô cho phép đánh giá sự tương tự giữa hai tập hợp dựa trên sự tương đồng giữa hai cấu trúc. Tiếp cận đánh giá dựa trên sự tương đồng cấu trúc chặt chẽ hơn so với tiếp cận dựa trên lực lượng của các tập hợp. Hơn nữa, cấu trúc tôpô được dùng là cấu trúc tôpô Hausdorff. Tại sao sử dụng cấu trúc tôpô này vì phương pháp chung để đánh giá độ quan trọng của thuộc tính đó là đánh giá sự phụ thuộc của thuộc tính quyết định. Trong đó, cấu trúc tôpô của thuộc tính quyết định là cấu trúc tôpô Hausdorff nên chương này đề xuất chọn lọc các thuộc tính điều kiện có cấu trúc tôpô Hausdorff.

3) Kích thước rút gọn từ thuật toán đề xuất.

Như đã đề cập bên trên, tiếp cận đánh giá thuộc tính theo cấu trúc tôpô chặt hơn so với các tiếp cận độ đo truyền thống. Theo các kết quả nghiên cứu của Yu và các cộng [39] đã chỉ ra hai phân hoạch khác nhau có thể có cùng cấu trúc tôpô. Do đó, tập rút gọn theo tiếp cận tôpô sẽ có kích thước nhỏ hơn so với tập rút gọn theo tiếp cận độ đo truyền thống. Hơn nữa, giai đoạn phân cụm các thuộc tính lại tiếp tục chia nhỏ tập thuộc tính ứng viên thành các nhóm thuộc tính con có cùng cấu trúc phụ thuộc. Đây là nguyên nhân chủ đạo ảnh hưởng tới việc cải thiện kích thước tập rút gọn với các thuật toán tốt nhất hiện nay.

4.5. Kết luận Chương 4

Chương 4, luận án trình bày về phương pháp rút gọn thuộc tính theo tiếp cận tôpô Hausdorff. Các đóng góp chính của Chương này gồm có:

- Đề xuất cấu trúc tôpô dựa trên quan hệ của các phép toán xấp xỉ trên không gian xấp xỉ mờ ngưỡng β;

- Đề xuất cấu trúc tôpô Hausdorff dựa trên định nghĩa tính phân biệt được của ma trận quan hệ mờ ngưỡng β;

- Đề xuất thuật toán tìm tập rút gọn dựa trên cấu trúc tôpô Hausdorff và định nghĩa khái niệm đồng cấu trúc phụ thuộc trong không gian tôpô Hausdorff.

Các kết quả thực nghiệm cho thấy thuật toán đề xuất là hoàn toàn vượt trội so với các phương pháp khác cả về thời gian thực hiện của thuật toán, độ chính xác phân lớp và kích thước của tập rút gọn thu được.

KẾT LUẬN

A. Những kết quả chính của luận án

Luận án tham gia vào luồng nghiên cứu về vấn đề tiền xử lý dữ liệu, ứng dụng cho các lĩnh vực khai thác dữ liệu và học máy với các lớp bài toán rút gọn thuộc tính cho bảng quyết định miền giá trị số. Trước tiên luận án tập trung khảo sát các vấn đề còn tồn tại của các phương pháp hiện có và đưa ra một số các đóng góp chính như sau:

Thứ nhất, luận án đề xuất *phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận tập thô mờ trực cảm*.

Nhận thấy các phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm gần đây còn chưa hiệu quả với các bộ dữ liệu nhiễu (các bộ dữ liệu có độ chính xác phân lớp ban đầu thấp), luận án đã nghiên cứu mối quan hệ giữa độ nhất quán của thuộc tính và độ chính xác phân lớp để xác định công thức xây dựng không gian xấp xỉ mờ trực cảm. Bên cạnh đó, với các ưu điểm của độ đo khoảng cách mờ, luận án đề xuất phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm với các bước chính như sau:

- Mở rộng độ đo khoảng cách mờ trên nền tập mờ trực cảm và xây dựng độ đo khoảng cách giữa các phân hoạch mờ trực cảm.

- Xây dựng độ đo đánh giá độ quan trọng của thuộc tính dựa trên khoảng cách giữa các phân hoạch mờ trực cảm, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm δ-equal để định nghĩa tập rút gọn và xây dựng điều kiện dừng của thuật toán

Trên cơ sở đó, luận án đề xuất phương pháp lai ghép filter - wrapper tìm tập rút gọn hiệu quả về độ chính xác phân lớp trên các tập dữ liệu nhiễu. Ngoài ứng dụng của độ đo đề xuất cho bài toán rút gọn thuộc tính, độ đo này có thể áp dụng cho một số bài toán phân lớp, dự báo, hỗ trợ ra quyết định có liên quan đến kĩ thuật tính toán

mềm trên tập các số mờ trực cảm.

Thứ hai, luận án đề xuất *phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận tôpô mờ trực cảm*.

Mặc dù phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm đề xuất đã cải thiện độ độ chính xác phân lớp của tập rút gọn trên các bộ dữ liệu nhiễu nhưng kích thước của tập rút gọn vẫn còn hạn chế. Bên cạnh đó, dựa trên các ưu điểm của cấu trúc tôpô và phần tử đơn vị của tôpô đại số, luận án đề xuất phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm với các bước chính như sau:

- Đề xuất công thức quan hệ ưu tiên mờ trực cảm để xây dựng không gian xấp xỉ mờ trực cảm.

- Đề xuất cấu trúc tôpô mờ trực cảm dựa trên các cơ sở và cơ sở con được định nghĩa theo không gian xấp xỉ mờ trực cảm.

- Đề xuất độ đo tương đồng giữa các tôpô dựa trên khoảng cách tương đồng giữa các cơ sở con, làm cơ sở để đề xuất phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm tôpô đơn vị, làm cơ sở để định nghĩa tập rút gọn và xây dựng điều kiện dừng của thuật toán.

Trên cơ sở đó, luận án đề xuất hai phương pháp tìm tập rút gọn như sau:

- Đề xuất phương pháp filter thuộc tính cho tập rút gọn hiệu quả về thời gian và kích thước.

- Đề xuất phương pháp lai ghép filter - wrapper kết hợp cấu trúc dữ liệu ngắn xếp (Stack) cho tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp.

Thứ ba, luận án đề xuất *phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận tôpô Hausdorff*.

Mặc dù phương pháp rút gọn thuộc tính theo tiếp cận tôpô mờ trực cảm cho tập rút gọn hiệu quả về kích thước và độ chính xác phân lớp. Tuy nhiên thời gian thực hiện còn chưa hiệu quả trên các bộ dữ liệu có số chiều và mẫu lớn. Bên cạnh đó, phương pháp chọn lọc thuộc tính của các phương pháp đề xuất hiện nay vẫn theo tiếp cận xây

dựng các độ đo, khó khả thi trên các tập dữ liệu lớn (big data). Hơn nữa, với các ưu điểm của cấu trúc tôpô Hausdorff, luận án đề xuất phương pháp rút gọn thuộc tính theo tiếp cận tôpô Hausdorff với các bước chính như sau:

- Xây dựng cấu trúc tôpô theo tiếp cận tập thô trên không gian xấp xỉ mờ trực cảm ngưỡng β.

- Đề xuất phương pháp xác định cấu trúc tôpô Hausdorff của thuộc tính, làm cơ sở để xây dựng phương pháp chọn lọc thuộc tính.

- Đề xuất khái niệm đồng cấu trúc thuộc tính, làm cơ sở để phân nhóm các thuộc tính có cấu trúc tôpô Hausdorff.

Trên cơ sở đó, luận án đề xuất phương pháp wrapper các nhóm thuộc tính cho tập rút gọn hiệu quả về thời gian và độ chính xác phân lớp.

Bên cạnh các kết quả nghiên cứu của luận án đã đạt được, những nghiên cứu này *vẫn còn tồn tại một số hạn chế như sau*:

- Đối với phương pháp rút gọn thuộc tính theo tiếp cận tập thô mờ trực cảm, luận án chưa xét đến các trường hợp bảng quyết định thay đổi về thuộc tính, về mẫu với các tình huống bổ xung, cập nhật và loại bỏ dữ liệu.

- Đối với phương pháp rút gọn thuộc tính theo tiếp cận tôpô đại số, luận án chưa xây dựng được các phép toán tính toán tôpô cho các nhóm thuộc tính.

C. Hướng phát triển tiếp theo của luận án

Thứ nhất, luận án cần hoàn thiện các vấn đề còn tồn tại để đáp ứng trong các trường hợp bảng quyết định lớn, bảng quyết định biến động. Trong đó, cần phát triển các công thức tính toán gia tăng, hoàn thiện cấu trúc đại số của tôpô với các phép toán hợp và giao của các cặp thuộc tính.

Thứ hai, hiện nay đã có nhiều phương pháp rút gọn thuộc tính trong bảng quyết định không đầy đủ theo tiếp cận mô hình tập thô mở rộng, tuy nhiên các kết quả nghiên cứu vẫn còn hạn chế về kích thước và độ chính xác phân lớp trên các tập rút gọn thu được. Do đó, hướng nghiên cứu thứ hai trong tương lai của luận án sẽ nhắm

tới rút gọn thuộc tính cho bảng quyết định không đầy đủ thông qua một số các hướng mở rộng cấu trúc tôpô theo tiếp cận tập thô như sau:

1) Mở rộng cấu trúc tôpô dựa trên không gian xấp xỉ của mô hình tập thô dung sai, nghiên cứu một số tính chất khả li nhằm tìm ra tiêu chuẩn chọn lọc thuộc tính và xây dựng điều kiện dừng của thuật toán.

2) Mở rộng cấu trúc tôpô dựa trên mối quan hệ của các phép toán xấp xỉ của mô hình tập thô dung sai, nghiên cứu một số tính chất khả li nhằm tìm ra tiêu chuẩn chọn lọc thuộc tính và xây dựng điều kiện dừng của thuật toán.

DANH MỤC CÁC CÔNG TRÌNH NGHIÊN CỨU

A. Các công trình đã công bố

[CT1] **Trần Thanh Đại**, Nguyễn Long Giang, Trần Thị Ngân, Hoàng Thị Minh Châu, "Rút gọn thuộc tính cho bảng quyết định đầy đủ theo tiếp cận Topo mờ", *Hội thảo quốc gia lần thứ XXIV: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, Thái Nguyên, 12/2021 pp. 318-325, 2021.

[CT2] **Trần Thanh Đại**, Nguyễn Long Giang, Trần Thị Ngân, Hoàng Thị Minh Châu, Vũ Thu Uyên, Vương Trung Hiếu, "Về một phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận topo mờ trực cảm", *Các công trình nghiên cứu và phát triển CNTT và truyền thông*, Hà Nội, số 2, tr. 57-64, 2022.

[CT3] Nguyen Truong Thang, Nguyen Long Giang, **Tran Thanh Dai**, Nguyen Trung Tuan, Nguyen Quang Huy, Pham Viet Anh, Vu Duc Thi, "A Novel Filter-Wrapper Algorithm on Intuitionistic Fuzzy Set for Attribute Reduction from Decision Tables", *International Journal of Data Warehousing and Mining (IJDWM)* , số 17(4), tr. 67-100, 2021. (SCIE Q4 IF 0.78).

[CT4] **Trần Thanh Đai**, Nguyễn Long Giang, Hoàng Thị Minh Châu, Trần Thị Ngân, "Rút gọn thuộc tính cho bảng quyết định theo tiếp cận tập thô mờ trực cảm", *Kỷ yếu Hội nghị Khoa học Công nghệ Quốc Gia lần thứ XIII: Nghiên cứu cơ bản và ứng dụng công nghệ thông tin*, Nha Trang, 10/2020, tr. 516-524, 2020.

[CT5] **Trần Thanh Đại**, Nguyễn Long Giang, Vũ Đức Thi, Phan Đăng Hưng,"Về một phương pháp rút gọn thuộc tính theo tiếp cận tôpô Hausdorff", *Hội thảo quốc gia lần thứ XXVI: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, Bắc Ninh, 10/2023,tr. 416-523, 2023.

B. Các công trình đang chờ phản biện

[CT6] **Tran Thanh Dai**, Nguyen Long Giang, Vu Duc Thi, Tran Thi Ngan, Hoang Thi Minh Chau, Le Hoang Son "A New Approach for Attribute Reduction from Decision Table based on Intuitionistic Fuzzy Topology", *Soft Computing*. (SCIE Q2 IF 3.8). Đang chờ phản biện vòng 2.

TÀI LIỆU THAM KHẢO

[1] Hồ Thị Phượng. *Phương pháp gia tăng rút gọn thuộc tính trong bảng quyết định thay đổi theo tiếp cận tập thô mờ*. Luận án Tiến sĩ Khoa học máy tính, Học viện Khoa học và Công nghệ-Viện Hàn lâm Khoa học và Công nghệ Việt Nam, 2021.

[2] Nguyễn Văn Thiện. *Rút gọn thuộc tính và trích lọc luật theo tiếp cận tập thô mờ*. Luận án Tiến sĩ Khoa học máy tính, Học viện Khoa học và Công nghệ-Viện Hàn lâm Khoa học và Công nghệ Việt Nam, 2018.

[3] Cao Chính Nghĩa. *Rút gọn thuộc tính trong bảng quyết định theo tiếp cận tập thô mờ*. Luận án Tiến sĩ Khoa học máy tính, Học viện Khoa học và Công nghệ-Viện Hàn lâm Khoa học và Công nghệ Việt Nam, 2014.

[4] Saba Bashir et al. "A Novel Feature Selection Method for Classification of Medical Data Using Filters, Wrappers, and Embedded Approaches". In: *Complexity* 2022 (2022), pp. 1–12.

[5] L. Meenachi and S. Ramakrishnan. "Differential evolution and ACO based global optimal feature selection with fuzzy rough set for cancer data classification". In: *Soft Computing* 24.24 (2020), pp. 18463–18475.

[6] Savita Ahlawat and Rahul Rishi. "A Genetic Algorithm Based Feature Selection for Handwritten Digit Recognition". In: *Recent Patents on Computer Science* 12.4 (2018), pp. 304–316.

[7] Hui huang Zhao et al. "Multiple classifiers fusion and CNN feature extraction for handwritten digits recognition". In: *Granular Computing* 5.3 (2020), pp. 411–418.

[8] Linhui Sun, Sheng Fu, and Fu Wang. “Decision tree SVM model with Fisher feature selection for speech emotion recognition”. In: *Eurasip Journal on Audio, Speech, and Music Processing* 2019.1 (2019).

[9] Serdar Yildirim, Yasin Kaya, and Fatih Kılıç. “A modified feature selection method based on metaheuristic algorithms for speech emotion recognition”. In: *Applied Acoustics* 173 (2021).

[10] Gunjan Ansari, Tanvir Ahmad, and Mohammad Najmud Doja. “Spam review classification using ensemble of global and local feature selectors”. In: *Cybernetics and Information Technologies* 18.4 (2018), pp. 29–42.

[11] Hekmat Mohammadzadeh and Farhad Soleimanian Gharehchopogh. “A novel hybrid whale optimization algorithm with flower pollination algorithm for feature selection: Case study Email spam detection”. In: *Computational Intelligence* 37.1 (2021), pp. 176–209.

[12] Antonio Jesús Fernández-García et al. “A recommender system for component-based applications using machine learning techniques”. In: *Knowledge-Based Systems* 164 (2019), pp. 68–84.

[13] B. Saravanan, V. Mohanraj, and J. Senthilkumar. “A fuzzy entropy technique for dimensionality reduction in recommender systems using deep learning”. In: *Soft Computing* 23.8 (2019), pp. 2575–2583.

[14] Zdzisław Pawlak. “Rough sets”. In: *International Journal of Computer & Information Sciences* 11.5 (1982), pp. 341–356.

[15] Anhui Tan et al. “Granularity and Entropy of Intuitionistic Fuzzy Information and Their Applications”. In: *IEEE Transactions on Cybernetics* 52.1 (2022), pp. 192–204.

[16] Baohua Liang, Lin Wang, and Yong Liu. “Attribute reduction based on improved information entropy”. In: *Journal of Intelligent and Fuzzy Systems* 36.1 (2019), pp. 709–718.

[17] Jiali He et al. “Attribute reduction in an incomplete categorical decision information system based on fuzzy rough sets”. In: *Artificial Intelligence Review* 55.7 (2022), pp. 5313–5348.

[18] Zdzislaw Pawlak, S. K.M. Wong, and Wojciech Ziarko. “Rough sets: probabilistic versus deterministic approach”. In: *International Journal of Man-Machine Studies* 29.1 (1988), pp. 81–95.

[19] Zdzislaw Pawlak. “Granularity of knowledge, indiscernibility and rough sets”. In: *1998 IEEE International Conference on Fuzzy Systems Proceedings - IEEE World Congress on Computational Intelligence*. Vol. 1. 1998.

[20] Zia Bashir et al. “The topological properties of intuitionistic fuzzy rough sets”. In: *Journal of Intelligent and Fuzzy Systems* 38.1 (2020), pp. 795–807.

[21] Yehai Xie and Xiuwei Gao. “Topological reduction algorithm for relation systems”. In: *Soft Computing* 26.22 (2022), pp. 11961–11971.

[22] Didier Dubois and Henri Prade. “Rough fuzzy sets and fuzzy rough sets”. In: *International Journal of General Systems* 17.2-3 (1990), pp. 191–209.

[23] Xiaoling Yang et al. “A noise-aware fuzzy rough set approach for feature selection”. In: *Knowledge-Based Systems* 250.109092 (2022), p. 109092.

[24] Zeyu Qiu and Hong Zhao. “A fuzzy rough set approach to hierarchical feature selection based on Hausdorff distance”. In: *Applied Intelligence* 52.10 (2022), pp. 11089–11102.

[25] Ramesh Kumar Huda and Haider Banka. “Efficient feature selection methods using PSO with fuzzy rough set as fitness function”. In: *Soft Computing* 26.5 (2022), pp. 2501–2521.

[26] Pei Liang et al. “Feature selection based on robust fuzzy rough sets using kernel-based similarity and relative classification uncertainty measures”. In: *Knowledge-Based Systems* 255.109795 (2022), p. 109795.

[27] Jin Ye et al. “A novel fuzzy rough set model with fuzzy neighborhood operators”. In: *Information Sciences* 544 (2021), pp. 266–297.

[28] Anil Kumar and P. S.V.S. Sai Prasad. “Incremental fuzzy rough sets based feature subset selection using fuzzy min-max neural network preprocessing”. In: *International Journal of Approximate Reasoning* 139 (2021), pp. 69–87.

[29] Shuang An, Qinghua Hu, and Changzhong Wang. “Probability granular distance-based fuzzy rough set model”. In: *Applied Soft Computing* 102 (2021).

[30] Zhaowen Li et al. “Entropy measurement for a hybrid information system with images: an application in attribute reduction”. In: *Soft Computing* 26.21 (2022), pp. 11243–11263.

[31] Jiucheng Xu et al. “Feature genes selection based on fuzzy neighborhood conditional entropy”. In: *Journal of Intelligent and Fuzzy Systems* 36.1 (2019), pp. 117–126.

[32] Pengfei Zhang et al. *Multi-source information fusion based on rough set theory: A review*. 2021.

[33] Nguyen Long Giang et al. “Novel Incremental Algorithms for Attribute Reduction from Dynamic Decision Tables Using Hybrid Filter-Wrapper with Fuzzy Partition Distance”. In: *IEEE Transactions on Fuzzy Systems* 28.5 (2020), pp. 858–873.

[34] Xiaohong Zhang, Bing Zhou, and Peng Li. “A general frame for intuitionistic fuzzy rough sets”. In: *Information Sciences* 216 (2012), pp. 34–49.

[35] Chris Cornelis, Martine De Cock, and Etienne E. Kerre. “Intuitionistic fuzzy rough sets: At the crossroads of imperfect knowledge”. In: *Expert Systems* 20.5 (2003), pp. 260–270.

[36] Anhui Tan et al. “Intuitionistic Fuzzy Rough Set-Based Granular Structures and Attribute Subset Selection”. In: *IEEE Transactions on Fuzzy Systems* 27.3 (2019), pp. 527–539.

[37] Zdzisław Pawlak. “Rough set approach to knowledge-based decision support”. In: *European Journal of Operational Research* 99.1 (1997), pp. 48–57.

[38] E. F. Lashin and T. Medhat. “Topological reduction of information systems”. In: *Chaos, Solitons and Fractals* 25.2 (2005), pp. 277–286.

[39] Hai Yu and Wan Rong Zhan. “On the topological properties of generalized rough sets”. In: *Information Sciences* 263 (2014), pp. 141–152.

[40] Keyun Qin and Zheng Pei. “On the topological properties of fuzzy rough sets”. In: *Fuzzy Sets and Systems* 151.3 (2005), pp. 601–613.

[41] Pankaj Kumar Singh and Surabhi Tiwari. “Topological structures in rough set theory: A survey”. In: *Hacettepe Journal of Mathematics and Statistics* 49.4 (2020), pp. 1270–1294.

[42] Qing E. Wu et al. “Topology theory on rough sets”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 38.1 (2008), pp. 68–77.

[43] Sang Eon Han. “Topological properties of locally finite covering rough sets and K-topological rough set structures”. In: *Soft Computing* 25.10 (2021).

[44] Mostafa K. El-Bably, Kamel K. Fleifel, and O. A. Embaby. “Topological approaches to rough approximations based on closure operators”. In: *Granular Computing* 7.1 (2022).

[45] Tareq M. Al-shami and Ibtesam Alshammari. “Rough sets models inspired by supra-topology structures”. In: *Artificial Intelligence Review* (2022).

[46] Chun Yong Wang. “Topological characterizations of generalized fuzzy rough sets”. In: *Fuzzy Sets and Systems* 312 (2017).

[47] Huishan Wu and Guilong Liu. “The relationships between topologies and generalized rough sets”. In: *International Journal of Approximate Reasoning* 119 (2020), pp. 313–324.

[48] Lei Zhou, Wei Zhi Wu, and Wen Xiu Zhang. “On intuitionistic fuzzy rough sets and their topological structures”. In: *International Journal of General Systems* 38.6 (2009), pp. 589–616.

[49] Zhi Pei, Daowu Pei, and Li Zheng. “Topology vs generalized rough sets”. In: *International Journal of Approximate Reasoning* 52.2 (2011), pp. 231–239.

[50] Zhengang Zhao. “On some types of covering rough sets from topological points of view”. In: *International Journal of Approximate Reasoning* 68 (2016), pp. 1–14.

[51] Wei Yao and Sang Eon Han. “A topological approach to rough sets from a granular computing perspective”. In: *Information Sciences* 627 (2023).

[52] Tareq M. Al-shami. “Topological approach to generate new rough set models”. In: *Complex and Intelligent Systems* 8.5 (2022), pp. 4101–4113.

[53] Jiucheng Xu et al. “Feature selection method for color image steganalysis based on fuzzy neighborhood conditional entropy”. In: *Applied Intelligence* 52.8 (2022), pp. 9388–9405.

[54] L. A. Zadeh. “Fuzzy sets”. In: *Information and Control* 8.3 (1965), pp. 338–353.

[55] Zhong Yuan et al. “Attribute reduction methods in fuzzy rough set theory: An overview, comparative experiments, and new directions”. In: *Applied Soft Computing* 107.107353 (2021), p. 107353.

[56] Krassimir T. Atanassov. “Intuitionistic fuzzy sets”. In: *Fuzzy Sets and Systems* 20.1 (1986), pp. 87–96.

[57] Seema Mishra and Rekha Srivastava. “Fuzzy topologies generated by fuzzy relations”. In: *Soft Computing* 22.2 (2018), pp. 373–385.

[58] Changzhong Wang et al. “Fuzzy rough set-based attribute reduction using distance measures”. In: *Knowledge-Based Systems* 164 (2019), pp. 205–212.

[59] Daren Yu, Qinghua Hu, and Congxin Wu. “Uncertainty measures for fuzzy relations and their applications”. In: *Applied Soft Computing Journal* 7.3 (2007).

[60] Qinghua Hu et al. “Fuzzy probabilistic approximation spaces and their information measures”. In: *IEEE Transactions on Fuzzy Systems* 14.2 (2006).

[61] Qinghua Hu et al. “Gaussian kernel based fuzzy rough sets: Model, uncertainty measures and applications”. In: *International Journal of Approximate Reasoning* 51.4 (2010).

[62] Jerzy W. Grzymala-Busse. “On the unknown attribute values in learning from examples”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 542 LNAI Part F2. 1991.

[63] Chris Cornelis, Richard Jensen, and Hurtado. “Attribute selection with fuzzy decision reducts”. In: *Information Sciences* 180.2 (2010).

[64] Qinghua Hu et al. “Kernelized fuzzy rough sets and their applications”. In: *IEEE Transactions on Knowledge and Data Engineering* 23.11 (2011).

[65] Changzhong Wang et al. “Fuzzy Rough Attribute Reduction for Categorical Data”. In: *IEEE Transactions on Fuzzy Systems* 28.5 (2020).

[66] Andrzej Skowron and Cecylia Rauszer. “The Discernibility Matrices and Functions in Information Systems”. In: *Intelligent Decision Support*. 1992.

[67] Eric C.C. Tsang et al. “Attributes reduction using fuzzy rough sets”. In: *IEEE Transactions on Fuzzy Systems* 16.5 (2008), pp. 1130–1141.

[68] Richard Jensen and Qiang Shen. “New approaches to fuzzy-rough feature selection”. In: *IEEE Transactions on Fuzzy Systems* 17.4 (2009).

[69] Chen Degang and Zhao Suyun. “Local reduction of decision system with fuzzy rough sets”. In: *Fuzzy Sets and Systems* 161.13 (2010).

[70] Degang Chen and Yanyan Yang. “Attribute reduction for heterogeneous data based on the combination of classical and fuzzy rough set models”. In: *IEEE Transactions on Fuzzy Systems* 22.5 (2014).

[71] Yanyan Yang et al. “Fuzzy rough set based incremental attribute reduction from dynamic data with sample arriving”. In: *Fuzzy Sets and Systems* 312 (2017).

[72] Jianhua Dai et al. “Maximal-discernibility-pair-based approach to attribute reduction in fuzzy rough sets”. In: *IEEE Transactions on Fuzzy Systems* 26.4 (2018), pp. 2174–2187.

[73] Yanyan Yang et al. “Incremental Perspective for Feature Selection Based on Fuzzy Rough Sets”. In: *IEEE Transactions on Fuzzy Systems* 26.3 (2018).

[74] Ye Liu et al. “Discernibility matrix based incremental feature selection on fused decision tables”. In: *International Journal of Approximate Reasoning* 118 (2020), pp. 1–26.

[75] Yizhu Li et al. “Accelerated multi-granularity reduction based on neighborhood rough sets”. In: *Applied Intelligence* 52.15 (2022), pp. 17636–17651.

[76] Meng Hu et al. “A novel approach to attribute reduction based on weighted neighborhood rough sets”. In: *Knowledge-Based Systems* 220.106908 (2021).

[77] Shuangjie Li et al. “Online streaming feature selection based on neighborhood rough set”. In: *Applied Soft Computing* 113.108025 (2021), p. 108025.

[78] Qinghua Hu et al. “Neighborhood rough set based heterogeneous feature subset selection”. In: *Information Sciences* 178.18 (2008).

[79] Changzhong Wang et al. “Feature selection based on maximal neighborhood discernibility”. In: *International Journal of Machine Learning and Cybernetics* 9.11 (2018).

[80] Qinghua Hu, Daren Yu, and Zongxia Xie. “Neighborhood classifiers”. In: *Expert Systems with Applications* 34.2 (2008), pp. 640–649.

[81] Xiaoling Yang et al. “Neighborhood rough sets with distance metric learning for feature selection[Formula presented]”. In: *Knowledge-Based Systems* 224.107076 (2021), p. 107076.

[82] Jinghua Liu et al. “ASFS: A novel streaming feature selection for multi-label data based on neighborhood rough set”. In: *Applied Intelligence* (2022).

[83] Wenhao Shu, Wenbin Qian, and Yonghong Xie. “Incremental feature selection for dynamic hybrid data using neighborhood rough set”. In: *Knowledge-Based Systems* 194.105516 (2020), p. 105516.

[84] Jihong Wan et al. “A novel hybrid feature selection method considering feature interaction in neighborhood rough set[Formula presented]”. In: *Knowledge-Based Systems* 227.107167 (2021), p. 107167.

[85] Dan Liu and Jingwei Li. “Safety monitoring data classification method based on wireless rough network of neighborhood rough sets”. In: *Safety Science* 118 (2019), pp. 282–296.

[86] Rachid Benouini et al. “Fast feature selection algorithm for neighborhood rough set model based on Bucket and Trie structures”. In: *Granular Computing* 5.3 (2020), pp. 329–347.

[87] Changzhong Wang et al. “Feature subset selection based on fuzzy neighborhood rough sets”. In: *Knowledge-Based Systems* 111 (2016), pp. 173–179.

[88] Binbin Sang et al. “Incremental Feature Selection Using a Conditional Entropy Based on Fuzzy Dominance Neighborhood Rough Sets”. In: *IEEE Transactions on Fuzzy Systems* 30.6 (2022), pp. 1683–1697.

[89] Panpan Chen, Menglei Lin, and Jinghua Liu. “Multi-Label Attribute Reduction Based on Variable Precision Fuzzy Neighborhood Rough Set”. In: *IEEE Access* 8 (2020), pp. 133565–133576.

[90] Kai Zhang, Jianming Zhan, and Wei Zhi Wu. “On Multicriteria Decision-Making Method Based on a Fuzzy Rough Set Model with Fuzzy α-Neighborhoods”. In: *IEEE Transactions on Fuzzy Systems* 29.9 (2021), pp. 2491–2505.

[91] Jiucheng Xu, Kaili Shen, and Lin Sun. “Multi-label feature selection based on fuzzy neighborhood rough sets”. In: *Complex and Intelligent Systems* 8.3 (2022), pp. 2105–2129.

[92] Binbin Sang et al. “Feature selection for dynamic interval-valued ordered data based on fuzzy dominance neighborhood rough set”. In: *Knowledge-Based Systems* 227.107223 (2021), p. 107223.

[93] Shivam Shreevastava, Anoop Kumar Tiwari, and Tanmoy Som. “Intuitionistic fuzzy neighborhood rough set model for feature selection”. In: *International Journal of Fuzzy System Applications* 7.2 (2018), pp. 75–84.

[94] Jingjing Xie, Bao Qing Hu, and Haibo Jiang. “A novel method to attribute reduction based on weighted neighborhood probabilistic rough sets”. In: *International Journal of Approximate Reasoning* 144 (2022), pp. 1–17.

[95] Xianyong Zhang, Jilin Yang, and Lingyu Tang. “Three-way class-specific attribute reducts from the information viewpoint”. In: *Information Sciences* 507 (2020), pp. 92–126.

[96] Yu Fang and Fan Min. “Cost-sensitive approximate attribute reduction with three-way decisions”. In: *International Journal of Approximate Reasoning* 104 (2019), pp. 112–139.

[97] Alireza Mansouri Ghroutkhar and Hassan Mishmast Nehi. “Fuzzy–rough set models and fuzzy-rough data reduction”. In: *Croatian Operational Research Review* 19.1 (2020), pp. 67–80.

[98] Xiao Zhang et al. “A fuzzy rough set-based feature selection method using representative instances”. In: *Knowledge-Based Systems* 151 (2018), pp. 216–229.

[99] Qinghua Hu, Daren Yu, and Zongxia Xie. “Information-preserving hybrid data reduction based on fuzzy-rough techniques”. In: *Pattern Recognition Letters* 27.5 (2006).

[100] Nguyen Ngoc Thuy and Sartra Wongthanavasu. “Hybrid filter–wrapper attribute selection with alpha-level fuzzy rough sets”. In: *Expert Systems with Applications* 193.116428 (2022), p. 116428.

[101] Yaojin Lin et al. “Attribute reduction for multi-label learning with fuzzy rough set”. In: *Knowledge-Based Systems* 152 (2018), pp. 51–61.

[102] Pankhuri Jain, Anoop Kumar Tiwari, and Tanmoy Som. “A fitting model based intuitionistic fuzzy rough feature selection”. In: *Engineering Applications of Artificial Intelligence* 89.103421 (2020), p. 103421.

[103] Zhiming Zhang and Jingfeng Tian. “On attribute reduction with intuitionistic fuzzy rough sets”. In: *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems* 20.1 (2012), pp. 59–76.

[104] Zhan ao Xue et al. “Variable precision multi-granulation covering rough intuitionistic fuzzy sets”. In: *Granular Computing* (2022).

[105] Anoop Kumar Tiwari et al. “An intuitionistic fuzzy-rough set model and its application to feature selection”. In: *Journal of Intelligent and Fuzzy Systems* 36.5 (2019), pp. 4969–4979.

[106] Bin Bin Sang, Xiao Yan Zhang, and Wei Hua Xu. “Attribute reduction of relative knowledge granularity in intuitionistic fuzzy ordered decision table”. In: *Filomat* 32.5 (2018), pp. 1727–1736.

[107] Zhiming Zhang. “Attributes reduction based on intuitionistic fuzzy rough sets”. In: *Journal of Intelligent and Fuzzy Systems* 30.2 (2016), pp. 1127–1137.

[108] Zhang Chuanchao. “Generalized dynamic attribute reduction based on similarity relation of intuitionistic fuzzy rough set”. In: *Journal of Intelligent and Fuzzy Systems* 39.5 (2020), pp. 7107–7122.

[109] Mohamadtaghi Rahimi et al. “An intuitionistic fuzzy entropy approach for supplier selection”. In: *Complex and Intelligent Systems* 7.4 (2021), pp. 1869–1876.

[110] M. B. Revanasiddappa and B. S. Harish. “A New Feature Selection Method based on Intuitionistic Fuzzy Entropy to Categorize Text Documents”. In: *International Journal of Interactive Multimedia and Artificial Intelligence* 5.3 (2018), p. 106.

[111] Pengfei Zhang et al. “Heterogeneous Feature Selection Based on Neighborhood Combination Entropy”. In: *IEEE Transactions on Neural Networks and Learning Systems* PP (2022), pp. 1–14.

[112] Zhong Yuan et al. “Fuzzy complementary entropy using hybrid-kernel function and its unsupervised attribute reduction”. In: *Knowledge-Based Systems* 231.107398 (2021), p. 107398.

[113] Thang Truong Nguyen et al. “A novel filter-wrapper algorithm on intuitionistic fuzzy set for attribute reduction from decision tables”. In: *International Journal of Data Warehousing and Mining* 17.4 (2021), pp. 67–100.

[114] H. I. Mustafa and O. A. Tantawy. “A new approach of attribute reduction of rough sets based on soft metric”. In: *Journal of Intelligent and Fuzzy Systems* 39.3 (2020), pp. 4473–4489.

[115] Bing Huang et al. “Distance-based Information Granularity and Hierarchical Structure for an Intuitionistic Fuzzy Granular Space”. In: *Fuzzy Information and Engineering* 8.2 (2016), pp. 147–168.

[116] Weiping Yang et al. “Distance measurement on intuitionistic fuzzy granular structure sets”. In: *Microsyst. Technol.* 27.4 (2019), pp. 1633–1639.

[117] Zhaowen Li, Tusheng Xie, and Qingguo Li. “Topological structure of generalized rough sets”. In: *Computers and Mathematics with Applications* 63.6 (2012), pp. 1066–1071.

[118] Zhen Ming Ma and Bao Qing Hu. “Topological and lattice structures of L-fuzzy rough sets determined by lower and upper sets”. In: *Information Sciences* 218 (2013), pp. 194–204.

[119] Zhiming Zhang. “Generalized intuitionistic fuzzy rough sets based on intuitionistic fuzzy coverings”. In: *Information Sciences* 198 (2012), pp. 186–206.

[120] Lingyun Yang and Luoshan Xu. “Topological properties of generalized approximation spaces”. In: *Information Sciences* 181.17 (2011).

[121] Sang Min Yun and Seok Jong Lee. “New approach to intuitionistic fuzzy rough sets”. In: *International Journal of Fuzzy Logic and Intelligent Systems* 20.2 (2020), pp. 129–137.

[122] Sang Min Yun, Yeon Seok Eom, and Seok Jong Lee. “Topology of the Redefined Intuitionistic Fuzzy Rough Sets”. In: *International Journal of Fuzzy Logic and Intelligent Systems* 21.4 (2021), pp. 369–377.

[123] William Zhu. “Topological approaches to covering rough sets”. In: *Information Sciences* 177.6 (2007), pp. 1499–1508.

[124] Lirun Su and William Zhu. “Dependence space of topology and its application to attribute reduction”. In: *International Journal of Machine Learning and Cybernetics* 9.4 (2018), pp. 691–698.

[125] Rehab Ali Ibrahim et al. “An improved runner-root algorithm for solving feature selection problems based on rough sets and neighborhood rough sets”. In: *Applied Soft Computing* 97.105517 (2020), p. 105517.

[126] Linghe Kong et al. “Distributed Feature Selection for Big Data Using Fuzzy Rough Sets”. In: *IEEE Transactions on Fuzzy Systems* 28.5 (2020), pp. 846–857.

[127] Mahendra Prasad, Sachin Tripathi, and Keshav Dahal. “An efficient feature selection based Bayesian and Rough set approach for intrusion detection”. In: *Applied Soft Computing Journal* 87.105980 (2020), p. 105980.

[128] Qiuna Zhang and Chunhai Hu. “Reduction Algorithm of Interval-Valued Intuitionistic Fuzzy Probability Rough Set Under Dominant Relation”. In: *International Journal of Pattern Recognition and Artificial Intelligence* (2022).

[129] Jingjing Yang, Qinghua Zhang, and Qin Xie. “Attribute reduction based on misclassification cost in variable precision rough set model”. In: *Journal of Intelligent and Fuzzy Systems* 37.4 (2019), pp. 5129–5142.

[130] Anhui Tan et al. “Reduction foundation with multigranulation rough sets using discernibility”. In: *Artificial Intelligence Review* 53.4 (2020), pp. 2425–2452.

[131] Qinghua Hu, Zongxia Xie, and Daren Yu. “Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation”. In: *Pattern Recognition* 40.12 (2007).