BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Phạm Minh Ngọc Hà

NGHIÊN CỨU MỘT SỐ PHƯƠNG PHÁP NÂNG CAO HIỆU QUẢ TÍNH TOÁN TẬP RÚT GỌN TRÊN KHÔNG GIAN XẤP XỈ MỜ

LUẬN ÁN TIẾN SĨ NGÀNH HỆ THỐNG THÔNG TIN

Hà Nội - Năm 2024

BỘ GIÁO DỤC
VÀ ĐÀO TẠO

VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ

Phạm Minh Ngọc Hà

NGHIÊN CỨU MỘT SỐ PHƯƠNG PHÁP NÂNG CAO HIỆU QUẢ TÍNH TOÁN TẬP RÚT GỌN TRÊN KHÔNG GIAN XẤP XỈ MỜ

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)*
	PGS. TS. Nguyễn Long Giang	**TS. Nguyễn Mạnh Hùng**

Hà Nội - Năm 2024

LỜI CAM ĐOAN

Tôi xin cam đoan toàn bộ nội dung của luận án này là công trình nghiên cứu của riêng tôi dưới sự hướng dẫn của PGS. TS Nguyễn Long Giang và TS. Nguyễn Mạnh Hùng tại Viện Công nghệ thông tin - Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Tất cả các kết quả nghiên cứu lý thuyết và thực nghiệm được trình bày trong luận án này đều là chính xác, trung thực và không sao chép từ bất kỳ nguồn tài liệu nào và dưới bất kỳ hình thức nào. Tôi xin cam đoan mọi thông tin được trích dẫn và sử dụng trong luận án này đều được ghi nguồn đầy đủ và tuân thủ các quy định về trích dẫn và sử dụng tài liệu. Tôi xin chịu trách nhiệm trước pháp luật về nội dung của luận án này.

Hà Nội, ngày ... tháng ... năm 2024

Phạm Minh Ngọc Hà

LỜI CẢM ƠN

Trong quá trình nghiên cứu và hoàn thiện luận án tiến sĩ này, tôi xin bày tỏ lòng biết ơn sâu sắc đến các Thầy Cô, các nhà nghiên cứu, các chuyên gia, và gia đình đã luôn đồng hành và hỗ trợ tôi suốt quãng thời gian dài làm nghiên cứu.

Đầu tiên, tôi xin bày tỏ lòng biết ơn chân thành đến các Cán bộ Phòng Đào tạo - Học viện Khoa học và Công nghệ, phòng quản lý Sau Đại học của Viện Công nghệ thông tin, đã dành sự quan tâm, chỉ bảo và hỗ trợ tận tình cho quá trình nghiên cứu của tôi.

Tôi xin gửi lời tri ân đặc biệt đến các Thầy hướng dẫn của tôi, PGS. TS Nguyễn Long Giang và TS. Nguyễn Mạnh Hùng, những người đã dành thời gian và công sức không ngừng để hướng dẫn, chỉ bảo và động viên tôi trong suốt quá trình thực hiện luận án này.

Tôi cũng muốn gửi lời cảm ơn đến các cộng sự, bạn bè, và gia đình đã luôn ở bên cạnh, động viên, và chia sẻ những ý kiến quý báu để giúp tôi vượt qua những khó khăn trong quá trình nghiên cứu.

Cuối cùng, tôi muốn gửi lời biết ơn sâu sắc đến gia đình, người thân, và những người yêu thương của tôi, đã luôn đồng lòng và hỗ trợ tôi không ngừng, mang lại sự ấm áp và động viên tinh thần cho tôi trong suốt chặng đường dài làm nghiên cứu.

Tôi xin kính chúc sức khỏe, hạnh phúc và thành công cho mọi người. Lời cảm ơn chân thành của tôi không thể nào diễn tả hết lòng biết ơn và tình cảm mà tôi muốn bày tỏ.

NCS. Phạm Minh Ngọc Hà

MỤC LỤC

LỜI CAM ĐOAN i
LỜI CẢM ƠN ii
MỤC LỤC vi
DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT vii
DANH MỤC CÁC KÝ HIỆU viii
DANH MỤC HÌNH VẼ x
DANH MỤC BẢNG BIỂU xi
MỞ ĐẦU 1

CHƯƠNG 1. TỔNG QUAN VỀ PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRÊN KHÔNG GIAN XẤP XỈ MỜ **9**

1.1 Mở đầu 9
1.1.1 Khái quát bài toán chọn lọc thuộc tính theo tiếp cận tập thô . . 9
1.1.2 Một số ứng dụng của bài toán chọn lọc thuộc tính 12
1.2 Các độ đo độ phụ thuộc của thuộc tính trên không gian xấp xỉ mờ . . . 14
1.2.1 Độ đo miền dương mờ 15
1.2.2 Độ đo Entropy mờ 15
1.2.3 Độ đo hạt thông tin 16
1.3 Phương pháp chọn lọc thuộc tính cải thiện độ chính xác trên không gian xấp xỉ mờ 17
1.3.1 Các nghiên cứu liên quan 17
1.3.2 Vấn đề còn tồn tại 18
1.4 Phương pháp chọn lọc thuộc tính gia tăng trên không gian xấp xỉ mờ . 20
1.4.1 Các nghiên cứu liên quan 20
1.4.2 Vấn đề còn tồn tại 22

1.5 Các kiến thức nền tảng 22
1.5.1 Tập thô truyền thống 22
1.5.2 Tập mờ truyền thống 25
1.5.3 Không gian xấp xỉ mờ 26
1.5.4 Tập thô mờ 28
1.6 Quy trình xây dựng và đánh giá thuật toán chọn lọc đặc trưng 29
1.6.1 Trình tự các bước xây dựng phương pháp chọn lọc đặc trưng . 29
1.6.2 Dữ liệu và chuẩn hóa dữ liệu 31
1.6.3 Mô hình phân lớp và phương pháp đánh giá 32
1.7 Kết luận Chương 1 35

CHƯƠNG 2. RÚT GỌN THUỘC TÍNH THEO TIẾP CẬN MỞ RỘNG TẬP THÔ MỜ ĐỘ CHÍNH XÁC THAY ĐỔI TRÊN KHÔNG GIAN XẤP XỈ MỜ 37

2.1 Các kiến thức cơ sở 38
2.1.1 Mô hình tập thô điều chỉnh 38
2.1.2 Mô hình tập thô mờ điều chỉnh 39
2.2 Đề xuất phương pháp cải tiến mô hình tập thô mờ điều chỉnh 40
2.2.1 Không gian xấp xỉ mờ đề xuất 41
2.2.2 Mô hình tập thô mờ điều chỉnh đề xuất 42
2.2.3 Độ đo độ phụ thuộc đề xuất 45
2.3 Đề xuất phương pháp rút gọn thuộc tính cái thiện độ chính xác phân lớp 46
2.3.1 Mô hình đề xuất 47
2.3.2 Thuật toán đề xuất 48
2.3.3 Ví dụ số minh họa thuật toán 49
2.4 Thực nghiệm và đánh giá 52
2.4.1 Kịch bản và môi trường thực nghiệm 52
2.4.2 Đánh giá thuật toán đề xuất 55

2.4.3 So sánh các thuật toán rút gọn thuộc tính cải thiện độ chính xác phân lớp 58
2.5 Kết luận Chương 2 62

CHƯƠNG 3. PHƯƠNG PHÁP TÍNH TOÁN GIA TĂNG TẬP RÚT GỌN THEO TIẾP CẬN TÍNH TOÁN HẠT THÔNG TIN TRÊN KHÔNG GIAN XẤP XỈ MỜ 63

3.1 Các kiến thức cơ sở 64
3.2 Độ đo hạt thông tin mờ 65
3.2.1 Hạt thông tin mờ 66
3.2.2 Độ đo hạt thông tin mờ 67
3.2.3 Xây dựng thuật toán rút gọn thuộc tính 67
3.2.4 Ví dụ số minh họa thuật toán 69
3.3 Xây dựng thuật toán cập nhật tập rút gọn khi bổ sung tập đối tượng . . 72
3.3.1 Công thức gia tăng khi bổ sung thêm một đối tượng 72
3.3.2 Công thức gia tăng khi bổ sung thêm một tập đối tượng 73
3.3.3 Đề xuất thuật toán gia tăng khi bổ sung tập đối tượng 75
3.4 Xây dựng thuật toán cập nhật tập rút gọn khi loại bỏ tập đối tượng . . 78
3.4.1 Công thức gia tăng khi loại bỏ một đối tượng 78
3.4.2 Công thức gia tăng khi loại bỏ một tập đối tượng 79
3.4.3 Đề xuất thuật toán gia tăng khi loại bỏ tập đối tượng 81
3.5 Thực nghiệm và đánh giá các thuật toán 84
3.5.1 Kịch bản và môi trường thực nghiệm 84
3.5.2 Đánh giá thuật toán FIG 85
3.5.3 Đánh giá thuật toán FIGAMO 89
3.6 Kết luận Chương 3 94

KẾT LUẬN 95

DANH MỤC CÁC CÔNG TRÌNH NGHIÊN CỨU 98

TÀI LIỆU THAM KHẢO **99**

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Thuật ngữ & chữ viết tắt	Diễn giải
AR	Rút gọn thuộc tính (Attribute Reduction)
FS	Chọn lọc thuộc tính (Feature Slection)
NDT	Bảng quyết định số đầy đủ
Reduct	Tập rút gọn
VPRS	Tập thô điều chỉnh (Variable Precision Rough Set)
VPFRS	Tập thô mờ điều chỉnh (Variable Precision Fuzzy Rough Set)
VPOFRS	Tập thô mờ điều chỉnh cải tiến
FD	Khảng cách mờ (Fuzzy Distance)
IFD	khoảng cách mờ trực cảm (Intuitionistic Fuzzy Distance)
FIG	Hạt thông tin mờ (Fuzzy Information Granular)
Raw	Bảng quyết định gốc
FIGAMO	Thuật toán gia tăng bổ sung tập đối tượng theo tiếp cận hạt thông tin mờ
FIGDMO	Thuật toán gia tăng loại bỏ tập đối tượng theo tiếp cận hạt thông tin mờ
FDMAO	Thuật toán gia tăng bổ sung tập đối tượng theo tiếp cận khoảng cách mờ
FDMDO	Thuật toán gia tăng loại bỏ tập đối tượng theo tiếp cận khoảng cách mờ
Granular	Hạt thông tin

DANH MỤC CÁC KÝ HIỆU

Ký hiệu	Diễn giải
C	Tập thuộc tính điều kiện
D	Tập thuộc tính quyết định
U	Tập đối tượng
$\mathscr{O}$	O lớn
$\mathbb{R}$	Tập số thực
M	Ma trận quan hệ của các đối tượng
R	Quan hệ tương đương của hai đối tượng
$\mathscr{R}$	Quan hệ tương đương mờ của hai đối tượng
$\|C\|$	Số lượng các thuộc tính điều kiện trong bảng quyết định
$\|D\|$	Số lượng các phân lớp của thuộc tính điều kiện
$\|D_k\|$	Số lượng các đối tượng của phân lớp k của thuộc tính điều kiện
$\|U\|$	Số lượng các đối tượng trong bảng quyết định
$\mathscr{F}(U)$	Họ các tập mờ trên U
$\underline{R}_\beta X$	Beta xấp xỉ dưới của X
$\bar{R}_\beta X$	Beta xấp xỉ trên của X
$\underline{\mathscr{R}}_\beta A(x)$	Beta thuộc của x trong tập xấp xỉ dưới của A
$\overline{\mathscr{R}_\beta} A(x)$	Beta thuộc của x trong tập xấp xỉ trên của A
$\gamma_\beta(C,D)$	Độ phụ thuộc của D vào C
$Sig_\beta(c,B)$	Độ quan trọng của thuộc tính c với tập thuộc tính B tại ngưỡng beta
$Sig^U_{D\|B}(a)$	Độ quan trọng của thuộc tính a trên U đối tượng
$Sig^{U\cup\Delta_u}_{D\|B}(a)$	Độ quan trọng của thuộc tính c khi bổ sung Δ_u đối tượng.

DANH MỤC CÁC HÌNH VẼ

1.1 Mô hình filter thuộc tính với chiến lược bổ sung thuộc tính 11
1.2 Mô hình filter thuộc tính với chiến lược loại bỏ thuộc tính 12
1.3 Mô hình wrapper thuộc tính với chiến lược tìm kiếm vét cạn 13
1.4 Mô hình lý thuyết tập thô 24

2.1 Mô hình lý thuyết tập thô mờ điều chỉnh 40
2.2 Mối quan hệ giữa kích thước tập rút gọn với ngưỡng β 54
2.3 Mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn trên mô hình SVM 55
2.4 Mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn trên mô hình k-NN 56
2.5 So sánh kích thước tập rút gọn thu được từ các thuật toán 59
2.6 So sánh độ chính xác phân lớp của các tập rút gọn xây dựng trên mô hình phân lớp SVM 60
2.7 So sánh độ chính xác phân lớp của các tập rút gọn xây dựng trên mô hình phân lớp k-NN 61
2.8 So sánh thời gian thực hiện giữa các thuật toán 62

3.1 Minh họa phương pháp gia tăng khi bổ sung một đối tượng 73
3.2 Minh họa phương pháp gia tăng khi loại bỏ một đối tượng 79
3.3 So sánh thời gian thực hiện giữa các thuật toán rút gọn thuộc tính . . . 86
3.4 So sánh kích thước tập rút gọn giữa các thuật toán rút gọn thuộc tính . 86
3.5 So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình SVM . 88
3.6 So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình KNN 88
3.7 So sánh thời gian thực hiện giữa các thuật toán khi bổ sung tập đối tượng 90

3.8 So sánh kích thước tập rút gọn giữa các thuật toán khi bổ sung tập đối tượng 91
3.9 So sánh độ chính xác phân lớp của các thuật toán trên mô hình SVM khi bổ sung tập đối tượng 92
3.10 So sánh độ chính xác phân lớp của các thuật toán trên mô hình KNN khi bổ sung tập đối tượng 93

DANH MỤC CÁC BẢNG BIỂU

1.1 Bảng thống kê các tiếp cận xây dựng độ đo miền dương 15
1.2 Bảng thống kê các tiếp cận xây dựng độ đo Entropy 16
1.3 Bảng thống kê các tiếp cận xây dựng độ đo hạt thông tin 16
1.4 Bảng thống kê các tiếp cận chọn lọc thuộc tính cải thiện độ chính xác phân lớp 19
1.5 Bảng thống kê phương pháp gia tăng khi thay đổi tập đối tương 21
1.6 Bảng thống kê phương pháp gia tăng khi thay đổi tập thuộc tính 22
1.7 Mô tả cấu trúc bảng quyết định số 25
1.8 Các toán tử cơ bản của tập mờ 26
1.9 Ma trận lầm lẫn nhị phân 33

2.1 Bảng mô tả các bộ dữ liệu có độ chính xác phân lớp thấp 53

3.1 Bảng mô tả dữ liệu gia tăng 85

MỞ ĐẦU

Tính cấp thiết của đề tài luận án

Chọn lọc thuộc tính, còn được gọi là chọn lọc đặc trưng, là một bước quan trọng trong phân tích dữ liệu và học máy thống kê. Quá trình này bao gồm việc lựa chọn một tập con các thuộc tính có liên quan từ tập thuộc tính ban đầu, sao cho thông tin quan trọng được giữ lại một cách tối đa. Chọn lọc thuộc tính mang lại nhiều lợi ích đáng kể: 1) giảm độ phức tạp tính toán, 2) cải thiện khả năng diễn giải mô hình, và 3) nâng cao khả năng dự đoán. Mục tiêu chính là tìm ra một tập con các đặc trưng, từ tập đặc trưng ban đầu, mà vẫn đảm bảo bảo toàn thông tin hoặc khả năng đưa ra quyết định chính xác. Các ứng dụng quan trọng của chọn lọc thuộc tính xuất hiện rộng rãi trong các lĩnh vực như nhận dạng mẫu và khai thác dữ liệu, bao gồm phân loại văn bản [1], [2], xử lý ảnh [3]–[5], và xử lý tiếng nói [6]–[9].

Năm 1982, Pawlak giới thiệu mô hình lý thuyết tập thô (Rough Set - RS) [10], được cộng đồng khoa học đánh giá cao về khả năng phân tích dữ liệu trong các tình huống không đầy đủ và thiếu nhất quán. Nhờ khả năng này, chọn lọc thuộc tính theo tiếp cận RS đã thu hút sự quan tâm của nhiều nhà nghiên cứu trong lĩnh vực lý thuyết tập thô trong nhiều năm qua [4], [11]–[16]. Dựa trên khái niệm không gian xấp xỉ (Approximation Space - AP) của RS, nhiều độ đo đã được đề xuất để định nghĩa reduct và hỗ trợ chọn lọc thuộc tính. Các nghiên cứu gần đây cho thấy rằng, các phương pháp rút gọn thuộc tính theo tiếp cận tập thô truyền thống mang lại nhiều kết quả đáng chú ý trong việc giảm số lượng thuộc tính mà vẫn bảo toàn khả năng phân lớp của bảng quyết định [1], [8], [14]–[17].

Tuy nhiên, tập thô truyền thống chủ yếu phù hợp với các bảng quyết định có miền giá trị rời rạc [18]. Do đó, cần phải rời rạc hóa dữ liệu của các bảng quyết định số (miền giá trị liên tục) trước khi thực hiện chọn lọc thuộc tính. Quá trình này phát sinh

thêm chi phí tính toán, có thể làm mất đi tính tự nhiên của dữ liệu, và tiềm ẩn nguy cơ làm mất thông tin quan trọng. Để khắc phục những hạn chế này, các nhà nghiên cứu đã đề xuất mở rộng RS trên không gian xấp xỉ mờ, tạo ra mô hình tập thô mờ (Fuzzy Rough Set - FRS) [19]–[21], và mở rộng RS trên không gian xấp xỉ mờ trực cảm, tạo ra mô hình tập thô mờ trực cảm (Intuitionistic Fuzzy Rough Set - IFRS) [22]. Các mô hình này cho phép xây dựng các phương pháp chọn lọc thuộc tính trực tiếp trên bảng quyết định gốc mà không cần rời rạc hóa.

Không gian xấp xỉ mờ của FRS sử dụng khái niệm quan hệ tương tự (similarity relation) thay cho quan hệ tương đương (equivalence relation) để xây dựng không gian quan hệ giữa các đối tượng. Nhờ đó, quan hệ giữa các đối tượng trong không gian xấp xỉ mờ trở nên mềm dẻo hơn so với quan hệ tương đương truyền thống. Mức độ quan hệ giữa các đối tượng được biểu diễn bằng các giá trị trong khoảng $[0, 1]$ thay vì chỉ 0 hoặc 1 như trong tập thô truyền thống. Hiện nay, việc phát triển các phương pháp chọn lọc thuộc tính dựa trên việc xây dựng độ đo trên không gian xấp xỉ mờ diễn ra rất sôi động, với nhiều độ đo điển hình đã được đề xuất, bao gồm độ đo FPOS [20], [21], [23]–[28], độ đo FIE [4], [29]–[32], và độ đo FD [6], [11], [33]–[35]. Tại Việt Nam, luận án của TS. Nguyễn Văn Thiện đã mở rộng một số độ đo trên không gian xấp xỉ mờ, rút gọn thuộc tính cho bảng quyết định số.

Tuy các nghiên cứu đã chỉ ra rằng các phương pháp chọn lọc thuộc tính theo tiếp cận xây dựng độ đo trên không gian xấp xỉ mờ của FRS hoạt động hiệu quả với các bộ dữ liệu có miền giá trị số, nhưng hiệu quả của chúng có thể giảm khi áp dụng cho các bộ dữ liệu số có độ nhạy cao về tỉ lệ phân nhầm lớp (tức là có nhiều nhiễu). Do đó, mô hình tập thô mờ độ chính xác thay đổi (Variable Precision Fuzzy Rough Sets - VPFRS) [19], [31], [36]–[50] và các độ đo xây dựng trên không gian xấp xỉ mờ trực cảm của mô hình IFRS [13], [20], [22], [41], [46], [51]–[54] đã được đề xuất để giải quyết vấn đề này.

Theo tiếp cận VPFRS, việc điều chỉnh thành phần trong tập xấp xỉ dưới sẽ ảnh hưởng đến miền dương của thuộc tính, dẫn đến độ phụ thuộc của thuộc tính sẽ thay

đổi. Khác với tiếp cận VPFRS, các độ đo xây dựng theo tiếp cận IFRS hoàn toàn phụ thuộc vào không gian xấp xỉ mờ trực cảm. Trong đó, mỗi phần tử của không gian xấp xỉ mờ trực cảm biểu diễn mức độ tương tự và không tương tự giữa hai đối tượng được xét. Do đó, không gian xấp xỉ mờ trực cảm mô tả mối quan hệ giữa các đối tượng đa chiều hơn so với không gian xấp xỉ mờ của FRS [52]. Các công trình nghiên cứu [51] cho thấy tiếp cận IFRS có thể cải thiện chất lượng reduct trên các bộ dữ liệu nhiễu, tuy nhiên thời gian tính toán còn nhiều hạn chế (chi phí tính toán gấp đôi tiếp cận FRS). Tại Việt Nam, luận án TS của tác giả Trần Thanh Đại đề xuất độ đo khoảng cách giữa các phân hoạch mờ trực cảm (Intuitionistic Fuzzy Distance - IFD), chọn lọc thuộc tính cho các bảng quyết định số có chứa nhiễu. Tuy nhiên thời gian tính toán còn hạn chế do việc xác định công thức tính độ thành viên và không thành viên cho AP. Do đó, *mục tiêu nghiên cứu thứ nhất* của luận án là nghiên cứu mở rộng mô hình VPFRS sao cho thời gian tính toán các tập xấp xỉ hiệu quả hơn mô hình VPFRS hiện có. Mục tiêu nghiên cứu thứ nhất thuộc *nhóm các phương pháp chọn lọc thuộc tính cho bảng quyết định tĩnh*, nghĩa là các bảng quyết định có nội dung không thay đổi theo thời gian.

Trong thực tế, các ứng dụng học máy thường xuyên phải cập nhật mô hình để thích ứng với những thay đổi của dữ liệu theo thời gian. Do đó, việc phát triển các phương pháp chọn lọc thuộc tính hiệu quả khi dữ liệu được cập nhật là một yêu cầu cấp thiết [55]. Đến nay, đã có nhiều phương pháp tính toán gia tăng được đề xuất nhằm cập nhật tập rút gọn một cách hiệu quả [3], [6], [55]–[92]. Kỹ thuật tính toán gia tăng này chỉ đánh giá các thông tin mới và kết hợp chúng với kết quả trước đó để cập nhật reduct. Có ba kịch bản thay đổi dữ liệu chính: thay đổi tập thuộc tính [55], [56], [58], [60], [61], thay đổi tập đối tượng [3], [6], [64], [69], [70], [74], [78]–[80], và thay đổi nội dung của đối tượg [3].

Tại Việt Nam, luận án tiến sĩ của Nguyễn Bá Quảng đã đề xuất một phương pháp tính toán gia tăng dựa trên độ đo khoảng cách, được xây dựng dựa trên tính đơn điệu của các phép hợp và giao giữa hai tập hợp. Gần đây, Yang và cộng sự đã đề xuất một

phương pháp tính toán gia tăng theo tiếp cận độ đo hạt thông tin tri thức [70]. Không giống như độ đo khoảng cách, độ đo hạt thông tin tri thức dựa trên độ thô và độ mịn của các phân hoạch, giúp công thức xây dựng đơn giản hơn và thời gian tính toán nhanh hơn. Tuy nhiên, nghiên cứu của Zhang và cộng sự mới chỉ phát triển độ đo hạt thông tin tri thức trên không gian xấp xỉ rõ (crisp approximation space), chứ chưa mở rộng nó trên không gian xấp xỉ mờ.

Do đó, *mục tiêu thứ hai* của luận án này là nghiên cứu và mở rộng độ đo hạt thông tin tri thức trên không gian xấp xỉ mờ, ứng dụng vào việc xây dựng một phương pháp cập nhật thuộc tính cho bảng quyết định số có sự thay đổi về đối tượng. Mục tiêu nghiên cứu thứ hai này thuộc *nhóm các phương pháp chọn lọc thuộc tính cho bảng quyết định có sự thay đổi về đối tượng*.

Mục tiêu nghiên cứu: Hạn chế chính của các phương pháp rút gọn thuộc tính hiện tại, áp dụng cho các bảng quyết định số có chứa nhiễu và các bảng quyết định động, là chi phí thời gian tính toán cao. Do đó, luận án này tập trung vào mục tiêu cải thiện thời gian tính toán tập rút gọn trên cả hai loại bảng quyết định này. Cụ thể, mục tiêu này được chia thành hai hướng nghiên cứu chính:

1. *Cải thiện thời gian tính toán tập rút gọn trên bảng quyết định số có chứa nhiễu*:

   Để đạt được mục tiêu này, luận án sẽ giải quyết các vấn đề nghiên cứu sau:

   - **Vấn đề 1:** Nghiên cứu tổng quan các phương pháp chọn lọc thuộc tính hiện có nhằm giảm thiểu ảnh hưởng của nhiễu. Phân tích ưu và nhược điểm của từng phương pháp, và lý giải tại sao luận án lựa chọn tiếp cận VPFRS (Variable Precision Fuzzy Rough Sets) để phát triển.
   - **Vấn đề 2:** Phát triển và tối ưu hóa các phép toán cơ bản, nhằm cải thiện hiệu quả thời gian tính toán các tập xấp xỉ trong VPFRS.
   - **Vấn đề 3:** Đề xuất một phương pháp chọn lọc thuộc tính mới, dựa trên tiếp cận VPFRS đã được mở rộng và tối ưu hóa.

2. *Cải thiện thời gian tính toán tập rút gọn trên các bảng quyết định động*:

   Để cải thiện thời gian tính toán tập rút gọn trên các bảng quyết định động nói chung, và đặc biệt là trên các bảng quyết định số có sự thay đổi về tập đối tượng, luận án sẽ tập trung vào các vấn đề sau:

   - **Vấn đề 1:** Nghiên cứu và đánh giá các phương pháp chọn lọc thuộc tính gia tăng hiện có, dựa trên tiếp cận tính toán hạt (granular computing). Xác định các khoảng trống nghiên cứu trong lĩnh vực này và lý giải tại sao luận án lựa chọn tiếp cận hạt thông tin tri thức (information-theoretic granular measure) để mở rộng.
   - **Vấn đề 2:** Mở rộng khái niệm hạt thông tin tri thức trên không gian xấp xỉ mờ và xây dựng một độ đo mới dựa trên tiếp cận tính toán hạt.
   - **Vấn đề 3:** Xây dựng các công thức tính toán gia tăng tương ứng với các trường hợp bổ sung và loại bỏ đối tượng trong bảng quyết định số.
   - **Vấn đề 4:** Đề xuất các phương pháp chọn lọc thuộc tính gia tăng mới, tương ứng với các công thức tính toán gia tăng đã được xây dựng.

Đối tượng nghiên cứu: Luận án tập trung vào việc rút gọn các bảng quyết định đầy đủ có miền giá trị số, thường được gọi chung là bảng quyết định số, thông qua hai nhóm phương pháp sau:

- Nhóm phương pháp chọn lọc thuộc tính nhằm cải thiện độ chính xác trong các bảng quyết định số có chứa nhiễu.
- Nhóm phương pháp chọn lọc thuộc tính gia tăng áp dụng cho các bảng quyết định số.

Để thực hiện nghiên cứu này, luận án dựa trên các kiến thức nền tảng sau:

- Tổng quan về các khái niệm cơ bản liên quan đến bảng quyết định số và định nghĩa reduct trong ngữ cảnh chọn lọc thuộc tính.

- Khảo sát lý thuyết tập thô (Rough Set) và các mở rộng của nó, cùng với các phương pháp chọn lọc thuộc tính dựa trên các lý thuyết này.
- Nghiên cứu quy trình chung để xây dựng reduct thông qua phương pháp chọn lọc thuộc tính.
- Tìm hiểu về các công cụ chuẩn hóa dữ liệu, cũng như các phương pháp đo lường và đánh giá hiệu quả của mô hình phân lớp dữ liệu.
- Nghiên cứu các bộ dữ liệu số đầy đủ (complete numerical datasets) có sẵn từ kho dữ liệu học máy UCI [93].

Phạm vi nghiên cứu: Luận án tập trung vào nghiên cứu các phương pháp chọn lọc thuộc tính dựa trên các biến thể của các độ đo được xây dựng trên không gian xấp xỉ mờ, với ứng dụng chính là chọn lọc thuộc tính cho hai trường hợp dữ liệu sau:

- **Phương pháp chọn lọc thuộc tính cải thiện độ chính xác** cho các bảng quyết định số có chứa nhiễu.
- **Phương pháp chọn lọc thuộc tính gia tăng** trên các bảng quyết định số có sự thay đổi về tập đối tượng.

Phương pháp nghiên cứu: Để đạt được kết quả nghiên cứu của luận án, nghiên cứu lý thuyết và nghiên cứu thực nghiệm được kết hợp nhằm đảm bảo tính chặt chẽ về cơ sở toán học và tính ứng dụng trên các bộ dữ liệu thực tiễn. Trong đó:

- *Về phương diện lý thuyết*: nghiên cứu và xây dựng các các định nghĩa, các mệnh đề được trình bày và chứng minh rõ ràng, chặt chẽ, đầy đủ dựa trên cơ sở lý thuyết vững chắc của tập thô và tập thô mở rộng.

- *Về góc độ thực nghiệm*: sử dụng các bộ dữ liệu mẫu từ UCI [93] là nguyên liệu đầu vào cho các thuật toán. Sử dụng các mô hình phân lớp dữ liệu tiêu chuẩn cho dữ liệu số, kết hợp với các độ đo và loại hình đánh giá nhằm kết luận khả năng phân lớp của reduct thu được.

Kết quả nghiên cứu: Hai kết quả nghiên cứu chính của luận án đã đạt được gồm có:

1. *Đề xuất phương pháp rút gọt thuộc tính theo tiếp cận mở rộng tập thô mờ độ chính xác thay đổi trong không gian xấp xỉ mờ*

   Nội dung chính của kết quả này được trình bày trong Chương 2 của luận án, với một số các điểm đóng góp chính như sau:

   - Đề xuất phương pháp mở rộng mô hình tập thô mờ độ chính xác thay đổi VPOFRS.
   - Đề xuất thuật toán rút gọn thuộc tính theo tiếp cận VPOFRS.

   Các kết quả phân tích và thực nghiệm cho thấy, thuật toán rút gọn thuộc tính đề xuất trên mô hình VPOFRS cho thời gian thực hiện nhanh hơn so với thuật toán rút gọn thuộc tính theo tiếp cận IFRS và VPFRS truyền thống. **Đáng chú ý, tập rút gọn thu được có kích thước nhỏ hơn và độ chính xác phân lớp cao hơn so với một số phương pháp hiện có trên một số bộ dữ liệu.** Các kết quả nghiên cứu đã được công bố trên các công trình [CT2, CT3] của luận án.

2. *Đề xuất phương pháp tính toán tập rút gọn gia tăng trên không gian xấp xỉ mờ*

   Nội dung chính của kết quả này được trình bày trong Chương 3 của luận án, với một số các điểm đóng góp chính như sau:

   - Đề xuất khái niệm hạt thông tin mờ và độ đo hạt thông tin mờ.
   - Đề xuất mô hình và thuật toán rút gọn thuộc tính theo tiếp cận hạt thông tin mờ.
   - Xây dựng công thức gia tăng khi bổ sung tập đối tượng và thuật toán cập nhật tập rút gọn tương ứng.
   - Xây dựng công thức gia tăng khi loại bỏ tập đối tượng và thuật toán cập nhật tập rút gọn tương ứng.

Các kết quả phân tích và thực nghiệm cho thấy, thuật toán rút gọn thuộc tính đề xuất theo tiếp cận độ đo hạt thông tin mờ cho thời gian thực hiện nhanh hơn so với thuật toán rút gọn thuộc tính theo tiếp cận khoảng cách mờ truyền thống. **Hơn nữa, tập rút gọn thu được có kích thước nhỏ hơn và độ chính xác phân lớp cao hơn so với một số phương pháp khác trên một số bộ dữ liệu**. Các kết quả nghiên cứu đã được công bố trên các công trình [CT1, CT4] của luận án.

Cấu trúc của luận án: Ngoài phần mở đầu và phần kết luận, luận án xây dựng 03 chương nghiên cứu với các nội dung chính như sau:

Chương 1. Nghiên cứu tổng quan về phương pháp rút gọn thuộc tính trên không gian xấp xỉ mờ. Chương này giới thiệu tổng quát về bài toán rút gọn thuộc tính, trình bày các kiến thức cơ sở của lý thuyết tập thô và tập thô mở rộng trên không gian xấp xỉ mờ, rút gọn thuộc tính cho bảng quyết định số.

Chương 2. Rút gọn thuộc tính theo tiếp cận mở rộng tập thô mờ độ chính xác thây đổi trên không gian xấp xỉ mờ. Chương này trình bày các đóng góp quan trọng trong việc xây dựng độ đo độ phụ thuộc mở rộng, có thời gian tính toán hiệu quả. Đáng chú ý, một số tập rút gọn trên các bảng quyết định số có chứa nhiễu có khả năng cải thiện độ chính xác phân lớp tốt hơn so với các phương pháp hiện có.

Chương 3 Phương pháp tính toán gia tăng tập rút gọn theo tiếp cận tính toán hạt trên không gian xấp xỉ mờ. Chương này trình bày các đóng góp chính trong việc xây dựng phương pháp tính toán gia tăng tập rút gọn hiệu quả trên không gian xấp xỉ mở. Hơn nữa, phương pháp đề xuất cũng trình bày một số kết quả thực nghiệm đáng chú ý về khả năng cải thiện độ chính xác phân lớp tốt trên một số bộ dữ liệu có kích thước lớn.

CHƯƠNG 1. TỔNG QUAN VỀ PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH TRÊN KHÔNG GIAN XẤP XỈ MỜ

Chương 1 của luận án cung cấp một cái nhìn tổng quan về bài toán chọn lọc thuộc tính và các kiến thức nền tảng cần thiết. Chương này trình bày tổng quan về các phương pháp chọn lọc thuộc tính trên không gian xấp xỉ mờ, tập trung vào hai hướng tiếp cận chính: cải thiện độ chính xác phân lớp trên các bảng quyết định số có chứa nhiễu và tính toán gia tăng trên các bảng quyết định số cập nhật tập các đối tượng.

Điểm nổi bật của chương là việc khảo sát và phân tích các nghiên cứu liên quan đến hai hướng tiếp cận này, chỉ ra các ưu và nhược điểm của các phương pháp hiện có, cũng như xác định các khoảng trống nghiên cứu. Cụ thể, chương này làm rõ các hạn chế về thời gian tính toán của các phương pháp cải thiện độ chính xác phân lớp của tập rút gọn thu được dựa trên các bảng quyết định số có chứa nhiễu như VPFRS và IFRS, đồng thời nêu bật sự cần thiết bài toán cập nhật mô hình khi dữ liệu thay đổi, là tiền đề xây dựng tiếp cận tính toán gia tăng tập rút gọn.

Bên cạnh đó, chương 1 cũng trình bày các kiến thức nền tảng về lý thuyết tập thô, tập mờ, không gian xấp xỉ mờ và các độ đo liên quan, cung cấp nền tảng lý thuyết cho các chương tiếp theo.

1.1. Mở đầu

1.1.1. Khái quát bài toán chọn lọc thuộc tính theo tiếp cận tập thô

Năm 1982, Pawlak đề xuất mô hình lý thuyết tập thô (Rough Set -RS) [10], được công đồng các nhà khoa học đánh giá cao về khả năng phân tích dữ liệu với các trường hợp dữ liệu: không đầy đủ, thiếu nhất quán. Dựa trên khả năng phân tích dữ liệu của RS, chọn lọc thuộc tính theo tiếp cận RS là ứng dụng được nhiều nhà nghiên cứu lý

thuyết tập thô quan tâm trong nhiều năm qua [4], [11]–[16]. Không giống như các phương pháp giảm chiều dữ liệu của PCA hay SVD, chọn lọc thuộc tính theo tiếp cận tập thô là quá trình chọn lọc thuộc tính để tạo ra một tập con của tập thuộc tính ban đầu sao cho bảo toàn thông tin, bảo toàn khả năng phân lớp so với tập dữ liệu gốc. Do đó, tính tự nhiên của dữ liệu sau khi rút gọn không bị thay đổi so với dữ liệu ban đầu.

Rút gọn thuộc tính còn được gọi là chọn lọc đặc trưng, là bước tiền xử lý quan trọng trong các tác vụ phân tích dữ liệu và học máy. Chọn lọc thuộc tính liên quan đến việc chọn một tập con các thuộc tính có liên quan từ tập thuộc tính ban đầu sao cho vẫn giữ lại được càng nhiều thông tin càng tốt. Quá trình này rất cần thiết vì nhiều lý do khác nhau như giảm độ phức tạp tính toán, cải thiện khả năng diễn giải mô hình và nâng cao khả năng dự đoán. Mục tiêu chính của việc lựa chọn đặc trưng là tìm ra một tập đặc trưng con, giữ lại thông tin quan trọng hoặc khả năng ra quyết định so với tập đặc trưng gốc. Quá trình chọn lọc thuộc tính thường bước qua các giai đoạn sau đây:

- *Đánh giá thuộc tính*: Đánh giá tầm quan trọng hoặc mức độ liên quan của từng thuộc tính trong tập dữ liệu. Có thể sử dụng nhiều kỹ thuật khác nhau như phân tích thống kê, phân tích tương quan và đo lường mức độ đóng góp thông tin của từng thuộc tính.

- *Lựa chọn thuộc tính*: Chọn một tập con các thuộc tính dựa trên tiêu chuẩn chọn lọc thuộc tính được sử dụng. Quá trình này có thể được thực hiện bằng nhiều chiến lược chọn lọc khác nhau bao gồm các phương pháp filter thuộc tính, phương pháp wrapper thuộc tính hoặc phương pháp nhúng thuộc tính. Chi tiết các phương pháp được mô tả trong các Hình 1.1, 1.2, 1.3.

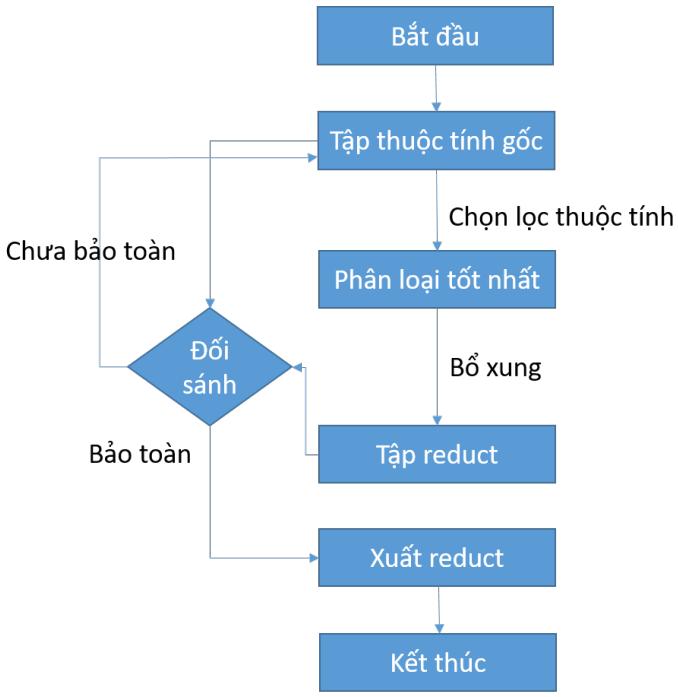
- *Xác thực tính đúng đắn của tập con*: Xác định tập con các thuộc tính đã chọn đã đảm bảo tính toàn vẹn thông tin so với tập thuộc tính gốc hay chưa. Tùy thuộc vào các phương pháp xây dựng thuật toán khác nhau mà phương pháp xác định tính chất đúng đắn của tập con cũng khác nhau. Khi đó, tập rút gọn theo từng phương pháp cũng sẽ được định nghĩa khác nhau.

- *Huấn luyện mô hình*: Huấn luyện mô hình phân lớp được chọn trên tập rút gọn

thu được và đánh giá hiệu năng phân loại của mô hình. Thông qua việc xây dựng mô hình trên tập rút gọn, mô hình có thể đạt được mức độ khái quát hóa tốt hơn, thời gian huấn luyện nhanh hơn và khả năng diễn giải của mô hình được cải thiện.

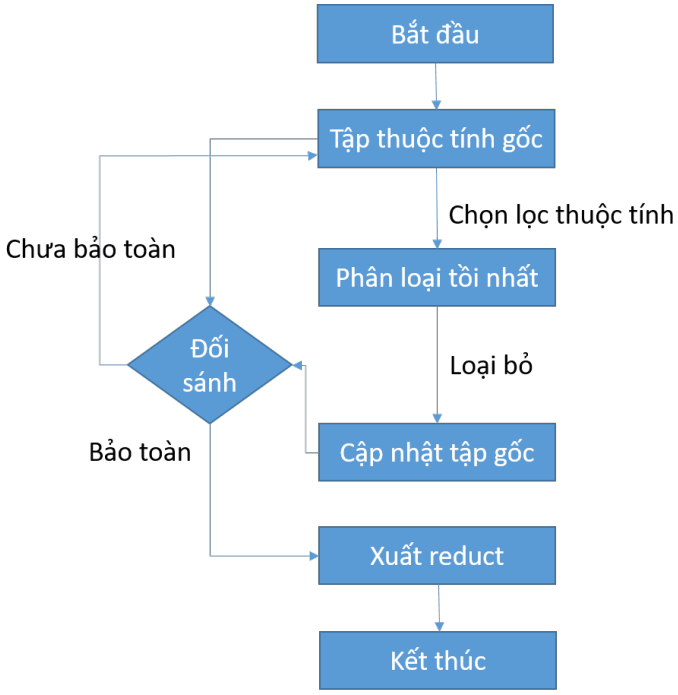
Dựa trên các phương pháp xác định tính chất đúng đắn của tập con, phương pháp chọn lọc thuộc tính, có ba phương pháp xây dựng thuật toán tìm tập con của tập thuộc tính theo các tiếp cận sau đây:

- Tiếp cận filter thuộc tính: tiếp cận này sử dụng các công cụ đo lường mức độ đóng góp thông tin của thuộc tính như: độ nhất quán, độ phụ thuộc, độ chắc chắn, độ kết hạt. Kết hợp với các chiến lược = bổ sung các thuộc tính quan trọng hoặc loại bỏ các thuộc tính dư thừa hoặc kết hợp cả hai để tìm tập thuộc tính rút gọn. Tiếp cận này cho thời gian tính toán là đa thức, phù hợp với năng lực tính toán của các thiết bị phần cứng hiện nay.



Hình 1.1: *Mô hình filter thuộc tính với chiến lược bổ sung thuộc tính*

- Tiếp cận wrapper thuộc tính: tiếp cận này sử dụng các phương pháp tìm kiếm tối ưu kết hợp với một bộ phân lớp dữ liệu để xác định tập đặc trưng con tối ưu nhất về khả năng phân lớp. Theo tiếp cận wrapper thuộc tính, số lượng các ứng viên reduct



Hình 1.2: *Mô hình filter thuộc tính với chiến lược loại bỏ thuộc tính*

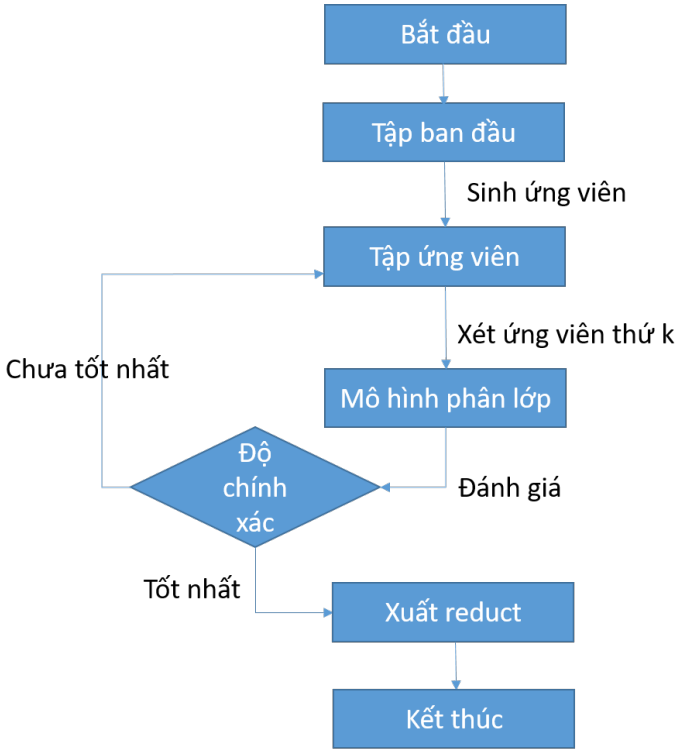
sinh ra là rất lớn, độ phức tạp tính toán là hàm mũ. Do đó, các thuật toán tìm kiếm tối ưu làm giảm không gian tìm kiếm, có thể đáp ứng được nhu cầu tính toán trên các bộ dữ liệu nhỏ. Tuy nhiên với các bộ dữ liệu có kích thước trung bình lớn hơn là không thể thực hiện được.

- Tiếp cận lai ghép filter-wrapper: Tiếp cận này khai thác những ưu điểm của hai phương pháp filter và wrapper, trong đó ưu điểm của filter là tính toán nhanh và ưu điểm của phương pháp wrapper là khả năng phân lớp chính xác cao.

1.1.2. Một số ứng dụng của bài toán chọn lọc thuộc tính

Các kỹ thuật chọn lọc thuộc tính trải rộng trên nhiều lĩnh vực và nhiều tác vụ khác nhau, bao gồm:

- Phân lớp dữ liệu: Lựa chọn thuộc tính thường được sử dụng để cải thiện hiệu suất của các mô hình phân loại bằng cách chọn các thuộc tính có tính phân biệt cao nhất và giảm nguy cơ khớp quá mức [1].



Hình 1.3: *Mô hình wrapper thuộc tính với chiến lược tìm kiếm vét cạn*

- Hồi quy dữ liệu: Trong phân tích hồi quy, việc giảm thuộc tính giúp xác định các yếu tố dự đoán có ảnh hưởng nhất và đơn giản hóa mô hình trong khi vẫn duy trì khả năng dự đoán của nó [2].

- Phân cụm dữ liệu: Lựa chọn thuộc tính có thể nâng cao thuật toán phân cụm bằng cách loại bỏ các thuộc tính không liên quan hoặc dư thừa, dẫn đến các cụm có ý nghĩa hơn [3].

- Giảm chiều: Giảm thuộc tính có liên quan chặt chẽ đến các kỹ thuật giảm kích thước như Phân tích thành phần chính (PCA) và Phân tích giá trị đơn lẻ (SVD), nhằm mục đích nắm bắt cấu trúc thiết yếu của dữ liệu trong không gian có chiều thấp hơn [4], [5].

- Phát hiện bất thường: Việc xác định hành vi bất thường hoặc các điểm bất thường trong tập dữ liệu có thể được hưởng lợi từ việc giảm thuộc tính để tập trung vào các thuộc tính mang lại nhiều thông tin nhất cho nhiệm vụ phát hiện bất thường [6]–[9].

Gần đây, một số ứng dụng bảo mật thông tin sử dụng phương pháp chọn lọc thuộc

tính để sàng lọc ra các đặc trưng gây hại nhất, các đặc trưng rủi ro nhất đến nguy cơ mất an toàn cho hệ thống thông tin của máy tính, gồm có:

- Phát hiện Spam Web: Tăng thứ hạng cho web thông qua tiếp cận spam là một hình thức đánh lừa các thuật toán của công cụ tìm kiếm. Vì vậy, các phương pháp khác nhau đã được đề xuất để phát hiện và cải thiện chất lượng kết quả tìm kiếm. Vì một trang web có thể được xem từ hai khía cạnh nội dung và liên kết, do đó số lượng trích dẫn có thể tăng rất cao. Vì vậy, việc chọn lọc các đặc trưng có khả năng phân loại cao là tiền xử lý dữ liệu quan trọng nhằm giảm thời gian và chi phí tính toán [94].

- Phát hiện Malware: Phần mềm độc hại là một trong những hình thức tấn công máy tính phổ biến nhất trong bối cảnh mối đe dọa ngày càng tăng của tội phạm mạng. Các nghiên cứu gần đây đã chứng minh rằng, phân loại được các hàm API có thể hiểu được chính xác hành vi của phần mềm độc hại. Do đó, việc chọn lọc hàm API nào có khả năng phân loại cao là hết sức quan trọng [95], [96].

- Phát hiện xâm nhập: Điện toán đám mây cung cấp cho người dùng các dịch vụ dựa trên Internet và một trong những thách thức chính đó là vấn đề bảo mật. Do đó, việc sử dụng Hệ thống phát hiện xâm nhập (IDS) làm chiến lược phòng thủ là điều cần thiết. Có nhiều ràng buộc trong một hệ thống IDS, khi đó việc lựa chọn những ràng buộc nào của IDS có khả năng phân loại tốt các hoạt động độc hại và hợp pháp là hết sức cần thiết [97].

1.2. Các độ đo độ phụ thuộc của thuộc tính trên không gian xấp xỉ mờ

Phần này sẽ khái quát các độ đo liên quan, ứng dụng xây dựng phương pháp chọn lọc thuộc tính trên không gian xấp xỉ mờ. Các độ đo này được xây dựng từ các biến thể mở rộng từ mô hình tập thô truyền thống trên không gian xấp xỉ mờ.

1.2.1. Độ đo miền dương mờ

Độ đo miền dương là độ đo dựa trên các giá trị thống kê được của các tập xấp xỉ dưới trong mô hình lý thuyết tập thô. Khi đó độ đo miền dương mờ là một độ đo được mở rộng từ độ đo miền dương cho không gian xấp xỉ mờ. Độ đo này thường được gọi là độ đo độ phụ thuộc miền dương, độ đo này thể hiện mức độ phụ thuộc của thuộc tính quyết định trên các thuộc tính điều kiện thông qua các giá trị khẳng định của miền dương thống kế được. Trong hệ thông tin nói chung và bảng quyết định nói riêng, độ đo này cho phép đo lường độ nhất quán. Một bảng quyết định được gọi là nhất quán nếu độ phụ thuộc của thuộc tính quyết định là lớn nhất (bằng 1). Chi tiết các nghiên cứu liên quan có thể được tìm thấy trong Bảng 1.1.

Bảng 1.1: *Bảng thống kê các tiếp cận xây dựng độ đo miền dương*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Mô hình	Không gian	Năm công bố
1	[23], [24]	Hỗn hợp	Tập thô mờ lân cận	Xấp xỉ mờ	2019, 2020
2	[25], [26]	Số	Tập thô mờ	Xấp xỉ mờ	2023
3	[20], [21]	Số	Tập thô mờ	Xấp xỉ mờ	2023
4	[27], [28]	Hỗn hợp	Tập thô mờ lân cận	Xấp xỉ mờ	2023

1.2.2. Độ đo Entropy mờ

Độ đo Entropy của Shanon là độ đo được sử dụng nhiều trong lĩnh vực học máy với cây quyết định. Độ đo này dùng để xác định số bít thông tin tối thiểu cần sử dụng để phân loại mỗi phần tử trong một tập. Khi đó, độ đo độ phụ thuộc của thuộc tính theo tiếp cận Entropy được xây dựng để đánh giá độ lợi thông tin của từng thuộc tính, làm tiêu chuẩn để xếp hạng mức độ quan trọng của các thuộc tính trong bảng quyết định. Trên môi trường không gian xấp xỉ mờ, độ đo Entropy cũng được mở rộng cho bài toán chọn lọc thuộc tính. Chi tiết các nghiên cứu liên quan có thể được tìm thấy trong Bảng 1.2.

Bảng 1.2: *Bảng thống kê các tiếp cận xây dựng độ đo Entropy*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Mô hình	Không gian	Năm công bố
1	[29]	Hỗn hợp	Entropy lân cận mờ	Xấp xỉ mờ	2019, 2020
2	[4]	Số	Entropy mờ	Xấp xỉ mờ	2023
3	[30]	Số	Entropy mờ	Xấp xỉ mờ	2023
4	[31], [32]	Hỗn hợp	Entropy lân cận mờ	Xấp xỉ mờ	2023

1.2.3. Độ đo hạt thông tin

Độ đo hạt thông tin là khái niệm quan trọng của lĩnh vực tính toán hạt. Trong môi trường tập thô truyền thống, độ đo hạt được xây dựng dựa trên các giá trị thống kê được của một phân hoạch hay một phủ. Độ đo hạt còn thể hiện mức độ thô và mịn của một phân hoạch, được gọi là độ thô và độ mịn của hạt. Khi đó, độ đo độ phụ thuộc theo tiếp cận tính toán hạt được xây dựng. Độ đo này được sử dụng để đánh giá độ phụ thuộc của thuộc tính trong bảng quyết định. Từ đó xây dựng độ đo đánh giá độ quan trọng của thuộc tính qua mức độ thô hay mịn của hạt thông tin phụ thuộc. Trên môi trường không gian xấp xỉ mờ, độ đo hạt thông tin cũng được mở rộng cho bài toán chọn lọc thuộc tính trên bảng quyết định số. Chi tiết các nghiên cứu liên quan có thể được tìm thấy trong Bảng 1.3.

Bảng 1.3: *Bảng thống kê các tiếp cận xây dựng độ đo hạt thông tin*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Mô hình	Không gian	Năm công bố
1	[11]	Hỗn hợp	Hạt thông tin lân cận mờ	Xấp xỉ mờ	2019, 2020
2	[6]	Số	Entropy mờ	Xấp xỉ mờ	2023
3	[33]	Số	Entropy mờ	Xấp xỉ mờ	2023
4	[34], [35]	Hỗn hợp	Hạt thông tin lân cận mờ	Xấp xỉ mờ	2023

1.3. Phương pháp chọn lọc thuộc tính cải thiện độ chính xác trên không gian xấp xỉ mờ

1.3.1. Các nghiên cứu liên quan

Sự phát triển của dữ liệu lớn (Big Data) đồng thời làm gia tăng các trường hợp dữ liệu thiếu, không đầy đủ và thiếu nhất quán. Các tập dữ liệu này thường cần được tiền xử lý trước khi đưa vào huấn luyện mô hình. Bên cạnh việc loại bỏ các thuộc tính dư thừa, việc xác định và loại bỏ các thuộc tính có độ nhất quán thấp (thuộc tính nhiễu) là một bước quan trọng. Việc loại bỏ các thuộc tính nhiễu không chỉ giúp giảm thời gian huấn luyện mô hình mà còn cải thiện đáng kể độ chính xác và độ tin cậy của mô hình.

Đối với các bảng quyết định có miền giá trị rời rạc và không đầy đủ, quan hệ giữa các đối tượng thường tạo thành các phủ (coverings). Do đó, các tập xấp xỉ được xây dựng dựa trên các phủ này thường không chính xác do sự không chắc chắn trong quan hệ giữa các đối tượng. Để giải quyết vấn đề này, Ziarko đã đề xuất phương pháp điều chỉnh thành phần trong tập xấp xỉ dựa trên mức độ thành viên của từng đối tượng, dẫn đến sự ra đời của mô hình Tập Thô Độ Chính Xác Thay Đổi (Variable Precision Rough Set - VPRS). Các nghiên cứu gần đây [10], [98] đã công bố các mô hình cải tiến và phương pháp chọn lọc thuộc tính cho bảng quyết định không đầy đủ, chứng minh khả năng cải thiện việc xử lý nhiễu trong các bảng quyết định rời rạc không đầy đủ.

Đối với bảng quyết định có miền giá trị số, mô hình Tập Thô Độ Chính Xác Thay Đổi được mở rộng trên không gian xấp xỉ mờ [31], [39], [45]. Trong không gian xấp xỉ mờ, các phép toán xấp xỉ dưới và xấp xỉ trên của tập thô mờ có thể được điều chỉnh để thay đổi độ chính xác của các tập xấp xỉ. Khi tập xấp xỉ dưới được điều chỉnh, độ phụ thuộc của thuộc tính trên miền dương cũng được cập nhật. Tuy nhiên, phương pháp thay đổi độ chính xác này phụ thuộc nhiều vào việc lựa chọn giá trị ngưỡng điều

chỉnh, và giá trị này thường khác nhau giữa các tập dữ liệu. Do đó, phương pháp này đòi hỏi nhiều kinh nghiệm và phân tích thống kê để chọn ngưỡng phù hợp.

Ngoài tiếp cận Tập Thô Mờ Độ Chính Xác Thay Đổi, tiếp cận sử dụng không gian xấp xỉ mờ trực cảm (Intuitionistic Fuzzy Rough Set - IFRS) cũng thu hút được sự quan tâm của nhiều nhà nghiên cứu [13], [20], [54]. Trong không gian xấp xỉ mờ trực cảm, các độ đo thường được xây dựng thông qua miền dương của IFRS hoặc trực tiếp trên không gian này. Khác với không gian xấp xỉ mờ truyền thống, không gian xấp xỉ mờ trực cảm có các ràng buộc chặt chẽ hơn dựa trên các hàm thành viên và không thành viên của tập mờ trực cảm. Tuy nhiên, việc xây dựng các hàm thành viên và không thành viên độc lập nhau là một thách thức lớn, thường đòi hỏi sử dụng một ngưỡng về độ do dự (hesitation degree) [54]. Hơn nữa, không gian lưu trữ và chi phí tính toán trên không gian xấp xỉ mờ trực cảm thường gấp đôi so với không gian xấp xỉ mờ truyền thống.

Do đó, luận án này đề xuất một hướng nghiên cứu mở rộng mô hình Tập Thô Mờ Độ Chính Xác Thay Đổi, ứng dụng vào bài toán chọn lọc thuộc tính để cải thiện độ chính xác trong các bảng quyết định số có chứa nhiễu. Chi tiết về các nghiên cứu liên quan có thể được tìm thấy trong Bảng 1.4.

1.3.2. Vấn đề còn tồn tại

Các nghiên cứu gần đây về phương pháp chọn lọc thuộc tính nhằm cải thiện độ chính xác phân lớp [31], [39], [45] đã chỉ ra rằng mô hình tập thô mờ với độ chính xác thay đổi, như được đề xuất bởi một số tác giả, còn hạn chế về hiệu quả thời gian tính toán. Cụ thể, phương pháp điều chỉnh của các tác giả này vẫn xem xét toàn bộ các đối tượng trong bảng quyết định, dẫn đến thời gian tính toán đáng kể. Trong khi đó, mô hình lý thuyết tập thô cho phép tận dụng một số đặc điểm quan trọng: tập các đối tượng trong một phân lớp luôn là tập con của tập đối tượng trong bảng quyết định, và các tính chất cơ bản của tập thô cho phép tùy biến các phép toán xấp xỉ phù hợp với nhiều trường hợp dữ liệu khác nhau. Vì vậy, luận án này đặt mục tiêu cải thiện thời

Bảng 1.4: *Bảng thống kê các tiếp cận chọn lọc thuộc tính cải thiện độ chính xác phân lớp*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Mô hình	Không gian	Năm công bố
1	[10], [98]	Hỗn hợp	Tập thô độ chính xác thay đổi	Xấp xỉ rõ	2022, 2023
2	[36], [42]	Số	Tập thô mờ độ chính xác thay đổi	Xấp xỉ mờ	2018, 2019
3	[19], [40], [43]	Số	Tập thô mờ độ chính xác thay đổi	Xấp xỉ mờ	2020, 2021
4	[41]	Số	Tập thô mờ độ chính xác thay đổi	Xấp xỉ mờ	2022
5	[31], [39], [45]	Số	Tập thô mờ độ chính xác thay đổi	Xấp xỉ mờ	2023
6	[51]	Số	Tập thô mờ trực cảm	Xấp xỉ mờ trực cảm	2020
7	[51], [53]	Số	Tập thô mờ trực cảm	Xấp xỉ mờ trực cảm	2020
8	[52], [54]	Số	Tập thô mờ trực cảm	Xấp xỉ mờ trực cảm	2021
9	[41]	Số	Tập thô mờ trực cảm	Xấp xỉ mờ trực cảm	2022
10	[13], [20]	Số	Tập thô mờ trực cảm	Xấp xỉ mờ trực cảm	2023

gian tính toán của quá trình chọn lọc thuộc tính, đặc biệt trong việc giảm thiểu ảnh hưởng của nhiễu, bằng cách tối ưu hóa các phép toán xấp xỉ dựa trên các tính chất cơ bản của tập thô mờ với độ chính xác thay đổi.

1.4. Phương pháp chọn lọc thuộc tính gia tăng trên không gian xấp xỉ mờ

Trong các bài toán yêu cầu cập nhật mô hình phân lớp liên tục, như các hệ thống giám sát giao thông, dữ liệu luôn được bổ sung và thay đổi. Điều này dẫn đến việc mô hình phân lớp cần được điều chỉnh theo các cập nhật dữ liệu. Trong bối cảnh đó, chọn lọc thuộc tính gia tăng đóng vai trò quan trọng trong việc giảm thời gian tiền xử lý dữ liệu, từ đó tăng hiệu quả về mặt thời gian cho quá trình cập nhật mô hình. Phần tiếp theo sẽ trình bày các nghiên cứu liên quan đến chọn lọc thuộc tính gia tăng, đồng thời thảo luận về những vấn đề còn tồn tại trong lĩnh vực này.

1.4.1. Các nghiên cứu liên quan

Các phương pháp chọn lọc thuộc tính dựa trên độ đo xây dựng từ tập xấp xỉ dưới thường bị giới hạn trong các bảng quyết định tĩnh, không cập nhật các mẫu mới. Tuy nhiên, trong thực tế, các bảng quyết định liên tục được cập nhật để nâng cao độ tin cậy của mô hình. Điều này đặt ra yêu cầu về khả năng tính toán hiệu quả, trong đó chỉ các thông tin mới được cập nhật cần được xử lý mà không cần tính toán lại toàn bộ dữ liệu. Do đó, tiếp cận tính toán gia tăng (incremental computation) [55], [56], [66], [74], [79] ngày càng thu hút sự chú ý của giới nghiên cứu. Kỹ thuật này chỉ đánh giá những thay đổi mới và kết hợp chúng với kết quả trước đó để tạo ra kết quả cập nhật, đáp ứng nhu cầu tối ưu hóa thời gian tính toán trong các ứng dụng thực tế. Các phương pháp rút gọn thuộc tính dựa trên tính toán gia tăng có thể được phân loại thành bốn nhóm, tương ứng với bốn loại thay đổi dữ liệu chính: thay đổi về tập đối tượng [3], [6], [64], [69], [70], [74], [78]–[80], thay đổi về tập thuộc tính [55], [56], [58], [60], [61], thay đổi về giá trị thuộc tính, và thay đổi hỗn hợp (kết hợp nhiều loại thay đổi). Thông tin chi tiết về các nghiên cứu liên quan được trình bày trong Bảng 1.5.

Trong trường hợp dữ liệu thay đổi về đối tượng, Jing và các cộng sự [3] đã trình bày về sự suy giảm độ quan trọng của thuộc tính khi tập đối tượng tăng lên và đề xuất

Bảng 1.5: *Bảng thống kê phương pháp gia tăng khi thay đổi tập đối tượng*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Độ đo	Năm công bố
1	[3], [69]	Số	Hạt thông tin	2018
2	[64]	Số	Hạt thông tin	2019
3	[6]	Số	Hạt thông tin	2020
4	[70], [78], [80]	Số	Hạt thông tin	2022
5	[74], [79]	Số	Hạt thông tin	2022
6	[76]	Số	Khoảng cách	2021

độ đo tương ứng. Liang và cộng sự [89] sử dụng entropy thông tin, trong khi Shu và các cộng sự đã sử dụng khái niệm phụ thuộc hàm để đề xuất phương pháp tính toán gia tăng trên các bảng quyết định động [59]. Yang và cộng sự đã sử dụng các kỹ thuật lấy mẫu để làm gọn các phần thông tin thay đổi [66], trong khi Zhang và cộng sự đã đề xuất một thuật toán gia tăng bằng kĩ thuật chọn cặp đại diện không nhất quán [79]. Giang và các cộng sự đề xuất một thuật toán gia tăng sử dụng khoảng cách mờ [76]. Liên quan đến sự biến đổi của các tập thuộc tính, Wang và các cộng sự [86] đã sử dụng entropy thông tin để tính toán mức độ suy giảm thông tin khi số lượng thuộc tính tăng lên, trong khi đó Jing và các cộng sự giới thiệu một thuật toán gia tăng cho bảng quyết định đầy đủ [3]. Niu và cộng sự trình bày phương pháp cập nhật tập rút gọn cho các bảng quyết định hỗn hợp [64], và Chen và các cộng sự [99] đã xác định mối quan hệ của các đối tượng khi thay đổi của các giá trị thuộc tính, Wang và các cộng sự [55] tính entropy đại diện, trong khi Jing và các cộng sự [3] đề xuất một phương pháp gia tăng dựa trên mực độ phân chia tri thức. Wei và các cộng sự xây dựng phương pháp tính toán gia tăng theo tiếp cận ma trận phân biệt [66]. Dong và Chen [55] trình bày phương pháp cập nhật tập rút gọn trong trường hợp dữ liệu biến động hỗn hợp. Jing các cộng sự đã áp dụng cách tiếp cận gia tăng để tối ưu thời gian cập nhật tập rút gọn trong trường hợp biến động cả về đối tượng và thuộc tính trong bảng quyết định [40].

Bảng 1.6: *Bảng thống kê phương pháp gia tăng khi thay đổi tập thuộc tính*

STT	Tài liệu tham chiếu	Kiểu dữ liệu	Độ đo	Năm công bố
1	[58], [61]	Số	Hạt thông tin	2018
2	[56], [60]	Số	Hạt thông tin	2019
3	[55]	Số	Hạt thông tin	2020

1.4.2. Vấn đề còn tồn tại

Kết quả khảo sát cho thấy rằng, trong bài toán cập nhật tập rút gọn khi bảng quyết định có sự thay đổi về tập đối tượng, độ đo dựa trên tiếp cận tính toán hạt (granular computing) tỏ ra hiệu quả trong việc xây dựng các công thức tính toán gia tăng. Bên cạnh đó, tiếp cận tính toán cũng cho thấy độ đo khoảng cách (distance measure) và độ đo hạt thông tin tri thức (information-theoretic granular measure) là hai công cụ hữu hiệu. Trong khi độ đo khoảng cách thường được xây dựng dựa trên các phép toán giao và hợp của tập hợp, độ đo hạt thông tin tri thức lại dựa trên độ thô (roughness) và độ mịn (granularity) của các phân hoạch. Nhờ vậy, các công thức tính toán gia tăng được xây dựng theo tiếp cận độ đo hạt thông tin tri thức có cấu trúc đơn giản hơn, dễ dàng chứng minh và xác nhận tính đúng đắn trong các trường hợp bổ sung hoặc loại bỏ đối tượng.

1.5. Các kiến thức nền tảng

1.5.1. Tập thô truyền thống

Mô hình Tập Thô, được đề xuất bởi Zdzisław Pawlak, là một phương pháp toán học để xử lý dữ liệu không đầy đủ, không chính xác và mơ hồ. Nó dựa trên việc xấp xỉ một tập hợp xác định (target set) bằng hai tập hợp khác: xấp xỉ dưới và xấp xỉ trên. Mục tiêu là tìm ra kiến thức tiềm ẩn trong dữ liệu mà không cần thêm bất kỳ thông tin trước nào.

Định nghĩa 1.1 (Hệ thống thông tin)**.** Một hệ thống thông tin được định nghĩa là một

bộ tứ $S = (U, A, V, f)$, trong đó:

- U là một tập hợp hữu hạn các đối tượng (còn gọi là vũ trụ - universe).
- A là một tập hợp hữu hạn các thuộc tính (attributes).
- $V = \bigcup_{a \in A} V_a$, với V_a là tập hợp các giá trị mà thuộc tính a có thể nhận.
- $f : U \times A \to V$ là một hàm gán giá trị cho mỗi thuộc tính của mỗi đối tượng, tức là $f(x,a) \in V_a$ với mọi $x \in U$ và $a \in A$.

Định nghĩa 1.2 (Quan hệ tương đương)**.** Cho $B \subseteq A$. Quan hệ tương đương $\mathrm{IND}(B)$ (indiscernibility relation) được định nghĩa như sau:

$$\mathrm{IND}(B) = \{(x,y) \in U \times U \mid \forall a \in B, f(x,a) = f(y,a)\} \tag{1.1}$$

Điều này có nghĩa là, hai đối tượng x và y là không phân biệt được (indiscernible) theo các thuộc tính trong tập B nếu và chỉ nếu chúng có cùng giá trị cho tất cả các thuộc tính đó.

Định nghĩa 1.3 (Lớp tương đương)**.** Tập hợp $[x]_B = \{y \in U \mid (x,y) \in \mathrm{IND}(B)\}$ là lớp tương đương chứa x dựa trên tập thuộc tính B.

Định nghĩa 1.4 (Xấp xỉ dưới)**.** Cho $X \subseteq U$. Xấp xỉ dưới của X theo B, ký hiệu là $\underline{B}X$, được định nghĩa như sau:

$$\underline{B}X = \{x \in U \mid [x]_B \subseteq X\} \tag{1.2}$$

Xấp xỉ dưới chứa tất cả các đối tượng mà chắc chắn thuộc về X dựa trên thông tin trong B.

Định nghĩa 1.5 (Xấp xỉ trên)**.** Cho $X \subseteq U$. Xấp xỉ trên của X theo B, ký hiệu là $\overline{B}X$, được định nghĩa như sau:

$$\overline{B}X = \{x \in U \mid [x]_B \cap X \neq \emptyset\} \tag{1.3}$$

Xấp xỉ trên chứa tất cả các đối tượng mà có thể thuộc về X dựa trên thông tin trong

B.

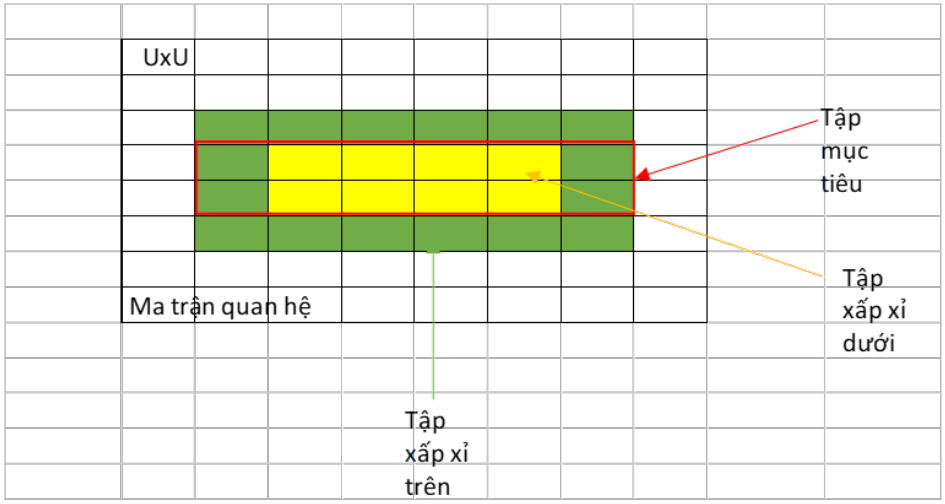
Định nghĩa 1.6 (Vùng biên)**.** Vùng biên của X theo B, ký hiệu là $\mathrm{BN}_B(X)$, được định nghĩa như sau:

$$\mathrm{BN}_B(X) = \overline{B}X - \underline{B}X \tag{1.4}$$

Vùng biên chứa các đối tượng mà không thể xác định chắc chắn là thuộc về X hay không thuộc về X dựa trên thông tin trong B.

Định nghĩa 1.7 (Tập hợp thô)**.** Tập hợp X được gọi là tập hợp thô nếu $\underline{B}X \neq \overline{B}X$. Nói cách khác, nếu vùng biên không rỗng, thì tập hợp đó là một tập hợp thô

Mô hình lý thuyết tập thô có thể được biểu diễn khái quát theo Hình 1.4



Hình 1.4: *Mô hình lý thuyết tập thô*

Định nghĩa 1.8. Bảng quyết định số được biểu diễn bởi bộ $NDT = (U, C, D, f)$. Trong đó U là tập hữu hạn khác rỗng các đối tượng, C là tập hữu hạn khác rỗng các thuộc tính điều kiện, D là tập hữu hạn các thuộc tính phân lớp và f là hàm xác định giá trị của mỗi đối tượng trong U tương ứng với các đặc trưng cho trước.

Bảng quyết định số 1.7 có $U = \{u_1, ..., u_6\}, C = \{a, b, c, d, e, f\}, D = \{D\}$. Trong đó $f(u_1, a) = 1.0, f(u_1, D) = 0$

Bảng 1.7: *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

1.5.2. Tập mờ truyền thống

Định nghĩa 1.9 (Khái niệm tập mờ)**.** Cho U là tập không rỗng, nếu $\mathscr{A}$ là một ánh xạ của U sang khoảng $[0,1]$, kí hiệu $\mathscr{A} : U \to [0,1]$ thì $\mathscr{A}$ được gọi là một tập mờ trên U. Với mọi $x_i \in U$, $A(x_i)$ được gọi là độ thuộc của x_i trong $\mathscr{A}$. Họ tất cả các tập mờ trên U được kí hiệu là $\mathscr{F}(U)$.

Định nghĩa 1.10 (Các tính chất cơ bản của tập mờ)**.** Cho $\mathscr{A}, \mathscr{B} \in \mathscr{F}(U)$. Với mọi $x \in U$:

$$\mathscr{A} = \mathscr{B} \leftrightarrow \mathscr{A}(x) = \mathscr{B}(x) \tag{1.5}$$

$$\mathscr{A} \subseteq \mathscr{B} \leftrightarrow \mathscr{A}(x) \leq \mathscr{B}(x) \tag{1.6}$$

$$\mathscr{A} \cup \mathscr{B}(x) = max\{\mathscr{A}(x), \mathscr{B}(x)\} \tag{1.7}$$

$$\mathscr{A} \cap \mathscr{B}(x) = min\{\mathscr{A}(x), \mathscr{B}(x)\} \tag{1.8}$$

$$\mathscr{A}^c(x) = 1 - \mathscr{A}(x) \tag{1.9}$$

Nhận thấy các phép toán hợp, giao và phép bù trên tập mờ là các phép toán mở rộng từ chuẩn tam giác ($t-norm$), đối chuẩn tam giác ($t-conorm$) và phép phủ định.

Xét $\mathscr{T} : [0,1] \times [0,1]$ là một ánh xạ. Với mọi $x, y, z \in [0,1]$, các điều kiện dau đây

được thỏa mãn:

(1) Tính giao hoán: $\mathscr{T}(x,y) = \mathscr{T}(y,x)$

(2) Tính kết hợp: $\mathscr{T}(x,\mathscr{T}(y,z)) = \mathscr{T}(\mathscr{T}(x,y),z)$

(3) Tính đơn điệu: $y \leq z \Rightarrow \mathscr{T}(x,y) \leq \mathscr{T}(x,z)$

(4) Điều kiện biên: $\mathscr{T}(x,1) = x$.

Khi đó $\mathscr{T}$ được gọi là chuẩn tam giác trên đoạn $[0,1]$. Nếu phép toán hai ngôi $\mathscr{S}$ thỏa mãn tính giao hoán, tính kết hợp, tính đơn điệu và $\mathscr{S}(x,0) = x$ thì $\mathscr{S}$ được gọi là phép toán đối chuẩn tam giác trên đoạn $[0,1]$. Phép toán phủ định $\mathscr{N}$ là một ánh xạ $[0,1] \to [0,1]$ không tăng trên đoạn $[0,1]$ thỏa mãn các điều kiện biên $\mathscr{N}(0) = 1$ và $\mathscr{N}(1) = 0$. Thông thường $\mathscr{N}(x) = 1 - x$ được gọi là phép toán phủ định tiêu chuẩn.

Cho $\mathscr{T}, \mathscr{S}$ và $\mathscr{N}$, ta nói $\mathscr{T}$ và $\mathscr{S}$ đối ngẫu qua $\mathscr{N}$ nếu $\mathscr{T}(x,y) = \mathscr{N}(\mathscr{S}(\mathscr{N}(x),\mathscr{N}(y)))$ và $\mathscr{S}(x,y) = \mathscr{N}(\mathscr{T}(\mathscr{N}(x),\mathscr{N}(y)))$. Cho $\mathscr{T}$ và $\mathscr{I} : \mathscr{I}(x,y) = sup\{z|\mathscr{T}(x,z) \leq y\}$. Khi đó $\mathscr{I}$ được goi là toán tử kéo theo mờ dựa trên $\mathscr{T}$. Tương tự $\theta(x,y) = inf\{z|\mathscr{S}(x,z) \geq y\}$ được gọi là toán tử kéo theo dựa trên $\mathscr{S}$ ($\mathscr{S}$-kéo theo). Một số phép toán $t-norm$, $t-conorm$ và kéo theo được trình bày trong bảng 1.8.

Bảng 1.8: *Các toán tử cơ bản của tập mờ*

chuẩn	đối chuẩn	kéo theo
$\mathscr{T}(x,y) = \min\{x,y\}$	$\mathscr{S}(x,y) = \max\{x,y\}$	$\mathscr{I}(x,y) = max(1-x,y)$
$\mathscr{T}(x,y) = xy$	$\mathscr{S}(x,y) = x+y-xy$	$\mathscr{I}(x,y) = 1-x+xy$
$\mathscr{T}(x,y) = \max\{x+y-1,0\}$	$\mathscr{S}(x,y) = \min\{x+y,1\}$	$\mathscr{I}(x,y) = min(1-x+y,1)$

1.5.3. Không gian xấp xỉ mờ

Định nghĩa 1.11 (Quan hệ tương tự của hai đối tượng trên một thuộc tính)**.** Cho bảng quyết định $NDT =< U,C,D >$, quan hệ tương tự của hai đối tượng $i, j \in U$ theo thuộc tính $a \in C$ được xác định bởi công thức sau:

$$s_{ij}^{a} = 1 - |a(i) - a(j)| \tag{1.10}$$

Rõ ràng công thức (1.6) có các tính chất sau:

(1) Tính phản xạ: $s^a_{ij} = 1$ nếu $i = j$;

(2) Tính đối tướng: $s^a_{ij} = s^a_{ji}$;

(3) Tính bắc cầu: $s^a_{ik} \geq min(s^a_{ij}, s^a_{jk})$.

Ví dụ: Cho bảng quyết định $NDT = (U, C, D)$ như bảng 1.7, xét thuộc tính $a \in C$ và hai đối tượng u_1, u_2. Khi đó quan hệ tương tự của hai đối tượng này được xác định bởi công thức (1.6) như sau: $s^a_{13} = 1 - |a(u_1) - a(u_2)| = 0.8$

Định nghĩa 1.12 (Ma trận quan hệ tương đương mờ của một thuộc tính)**.** Cho bảng quyết định $NDT =< U, C, D >$, quan hệ tương tự giữa các đối tượng của U theo thuộc tính $a \in C \cup D$ có thể được biểu diễn bởi ma trận quan hệ mờ như sau:

$$S(a,U) = \begin{bmatrix} s^a_{11} & s^a_{12} & \dots & s^a_{1n} \\ s^a_{21} & s^a_{22} & \dots & s^a_{2n} \\ \dots & \dots & \dots & \dots \\ s^a_{n1} & s^a_{n2} & \dots & s^a_{nn} \end{bmatrix} \tag{1.11}$$

Trong đó, $n = |U|$, $s^a_{ij} \in [0,1] \forall i, j \in [1,n]$ là quan hệ tương tự giữa đối tượng i và đối tượng j trong U trên thuộc tính a.

Ví dụ: Cho bảng quyết định $NDT = (U, C, D)$ như bảng 1.7, xét thuộc tính $a \in C$. Khi đó ma trận quan hệ tương tự của các đối tượng trong U theo thuộc tính $a \in C$ được biểu diễn qua ma trận (1.7) như sau: $R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}$.

Định nghĩa 1.13 (Ma trận quan hệ tương đương mờ của một tập thuộc tính)**.** Cho bảng quyết định $NDT =< U, C, D >$, ma trận quan hệ của các đối tượng trong U trên tập

thuộc tính $P \subseteq C$ được xác định như sau:

$$S(P,U) = \bigcap_{a \in P} S(a,U) \tag{1.12}$$

Trong đó, $s_{ij}^{P} = min\{s_{ij}^{a} : a \in P\}$.

Ví dụ: Xét hai ma trận quan hệ của các đối tượng trong U tương ứng với hai thuộc tính $a, b \in C$ như sau:

$$R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}; R_b = \begin{bmatrix} 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \end{bmatrix}.$$

Khi đó ma trận quan hệ của tập thuộc tính a, b được xác định bởi công thức (1.8) là: $R_{ab} = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 0.8 & 0.8 \\ 0.2 & 0.2 & 0.4 & 0.8 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 0.8 & 1 & 1 \end{bmatrix}$

1.5.4. Tập thô mờ

Cho $\mathscr{R} : U \times U \to [0,1]$ là quan hệ mờ từ U sang U. Khi đó, với mọi $x, y \in U$, $\mathscr{R}(x,y)$ xác định mức độ tương tự của x và y thông qua quan hệ $\mathscr{R}$. Quan hệ $\mathscr{R}$ của các đối tượng trên U có thể được biểu diễn bởi ma trận quan hệ mờ $M_{\mathscr{R}} = [r_{ij}]_{n \times n}$, với $r_{ij} = \mathscr{R}(x_i, x_j)$ và $n = |U|$. Khi đó, mỗi hàng của ma trận là một vector quan hệ mờ hay là một tập mờ trên U. Kí hiệu tập các quan hệ của $\mathscr{R}$ từ U sang U là $\mathscr{F}(U \times U)$.

Định nghĩa 1.14 (Quan hệ tương đương mờ)**.** Xét $\mathscr{R} \in \mathscr{F}(U \times U)$. Với mọi $x, y \in U$, $\mathscr{R}$ được gọi là quan hệ tương đương mờ nếu:

(1) $\mathscr{R}(x,x) = 1$: có tính phản xạ;

(2) $\mathscr{R}(x,y) = \mathscr{R}(y,x)$: có tính đối xứng;

(3) $\mathscr{R}(x,z) \geq min\{\mathscr{R}(x,y), \mathscr{R}(y,z)\}$: có tính bắc cầu - *min*.

Định nghĩa 1.15 (Tập thô mờ)**.** Cho bảng quyết định $NDT = (U, C, D, f)$, $\mathscr{R}$ là quan hệ tương đương mờ trên U. Với mọi $\mathscr{A} \in \mathscr{F}(U)$, độ thuộc của đối tượng $x \in U$ vào tập xấp xỉ dưới và xấp xỉ trên của A theo quan hệ $\mathscr{R}$ lần lượt được xác định như sau:

$$\underline{\mathscr{R}}\mathscr{A}(x) = \inf_{y \in U} \mathscr{I}(\mathscr{R}(x,y), \mathscr{A}(y)) \tag{1.13}$$

$$\overline{\mathscr{R}}\mathscr{A}(x) = \sup_{y \in U} \min(\mathscr{R}(x,y), \mathscr{A}(y)) \tag{1.14}$$

Ví dụ: Xét ma trận quan hệ $R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}$ và lớp mục tiêu $A = \{u_1, u_3, u_6\} = [1,0,1,0,0,1]$. Khi đó, độ thuộc của đối tượng u_1 với tập mục tiêu A được xác định bởi công thức (1.9) như sau:

$= \min\{\max(1-1,1), \max(1-1,0), ..., \max(1-0.2,1)\}$

$= \min\{1,0,1,0.8,0.8,0.8\} = 0$

1.6. Quy trình xây dựng và đánh giá thuật toán chọn lọc đặc trưng

1.6.1. Trình tự các bước xây dựng phương pháp chọn lọc đặc trưng

Để xây dựng phương pháp chọn lọc đặc trưng, hầu hết các phương pháp chọn lọc đặc trưng theo tiếp cận tập thô truyền thống và tập thô mở rộng đều xây dựng theo lộ trình các bước sau đây:

- Xây dựng phương pháp đánh giá, phân loại và chọn lọc thuộc tính: tiếp cận ma trận phân biệt có thể phân loại các thuộc tính theo mức độ phân biệt của các đối tượng, tiếp cận độ đo có thể đánh giá mức độ phụ thuộc, lượng thông tin và kích thước hạt thông tin của thuộc tính.

- Xây dựng tập rút gọn thông qua các điều kiện cần (tính bảo toàn) và điều kiện đủ (tính tối thiểu) được định nghĩa.

- Thiết kế thuật toán: sử dụng phương pháp filter thuộc tính với các chiến lược bổ sung các thuộc tính quan trọng hoặc loại bỏ các thuộc tính dư thừa, hoặc kết hợp cả hai, sử dụng phương pháp wrapper kết hợp với các thuật toán tính toán tiến hóa. Kết hợp phương pháp filter và phương pháp wrapper thuộc tính cho tập rút gọn có chất lượng phân lớp tốt và thời gian thực hiện của thuật toán có bậc đa thức. *Thuật toán 1.1* là là mô hình tổng quát được sử dụng để xây dưng cho hầu hết các thuật toán rút gọn thuộc tính.

Thuật toán 1.1 Thuật toán QuickReduct

Đầu vào: Tập thuộc tính điều kiện A
Đầu ra: Tập rút gọn B

```
1:  B ← ∅;
2:  while M(B) ≠ M(A) do
3:      T ← B;
4:      best ← −1;
5:      for all a ∈ (A − B) do
6:          if M(B ∪ a) ≥ best then
7:              T ← B ∪ a;
8:              best ← M(B ∪ a);
9:          end if
10:     end for
11:     B ← T;
12: end while
13: return B
```

- Cuối cùng là bước đánh giá tính hiệu quả của phương pháp đề xuất dựa trên các tiêu chí hoặc cặp tiêu chí về độ chính xác phân lớp và kích thước của tập rút gọn, thời gian thực hiện của thuật toán.

Trong đó, M là một độ đo (Metric) nhằm đánh giá độ phụ thuộc, lượng thông tin hay hạt thông tin của tập đặc trưng. Ta có B sẽ là tập rút gọn của A nếu $M(B) = M(A)$.

1.6.2. Dữ liệu và chuẩn hóa dữ liệu

Dữ liệu thử nghiệm trong các Chương nghiên cứu của luận án được tải về từ kho dữ liệu UCI. Đây là tập hợp các bộ dữ liệu cho học máy được cung cấp miễn phí, luôn được cập nhật bởi Đại học California, Irvine. Những bộ dữ liệu này bao gồm nhiều chủ đề khác nhau và thường được sử dụng cho lĩnh vực nghiên cứu và giáo dục. Trước khi xây dựng các mô hình học máy, dữ liệu cần được chuẩn hóa. Trong đó, chuẩn hóa dữ liệu trong học máy và khai thác dữ liệu là quá trình điều chỉnh và biến đổi dữ liệu sao cho chúng phù hợp và dễ dàng sử dụng trong các mô hình học máy và quy trình khai thác dữ liệu. Một số tác vụ chuẩn hóa dữ liệu cơ bản gồm có:

- Chuẩn hóa đơn vị: Chuyển đổi đơn vị của các thuộc tính trong dữ liệu về cùng một miền giá trị hoặc đơn vị đo lường. Ví dụ, chuyển đổi các đơn vị đo lường từ mét sang centimet hoặc từ đồng sang USD.

- Chuẩn hóa phân phối: Đảm bảo phân phối của dữ liệu là đồng nhất bằng cách chuyển đổi các biến số sao cho chúng tuân theo phân phối chuẩn hoặc phân phối Gaussian.

- Chuẩn hóa định dạng: Đảm bảo dữ liệu được biểu diễn ở định dạng thích hợp cho mô hình học máy hoặc công cụ khai thác dữ liệu cụ thể. Ví dụ, chuyển đổi văn bản thành vectơ đặc trưng hoặc chuyển đổi dữ liệu hình ảnh thành định dạng số.

- Loại bỏ nhiễu: Loại bỏ các giá trị nhiễu, dữ liệu bị thiếu hoặc các giá trị ngoại lai có thể ảnh hưởng đến hiệu suất của mô hình học máy hoặc quá trình khai thác dữ liệu.

- Chuẩn hóa thời gian: Nếu dữ liệu có thông tin thời gian cần được chuẩn hóa và đồng bộ hóa sao cho các mô hình có thể hiểu được và sử dụng hiệu quả.

- Chuẩn hóa tỷ lệ: Đảm bảo các biến số có cùng tỷ lệ, tránh tình trạng biến số có giá trị lớn hơn có ảnh hưởng lớn hơn đến mô hình học máy so với biến số có giá trị

nhỏ hơn.

Trong quá trình khai thác dữ liệu và xây dựng mô hình học máy, việc chuẩn hóa dữ liệu đóng một vai trò quan trọng nhằm bảo đảm hiệu suất và độ chính xác của mô hình. Đồng thời, việc hiểu rõ về loại dữ liệu cũng giúp cho quá trình chuẩn hóa được thực hiện một cách hiệu quả và chính xác.

1.6.3. Mô hình phân lớp và phương pháp đánh giá

1.6.3.1. Mô hình đánh phân lớp

Mô hình phân loại SVM (Support Vector Machine) và kNN (k-Nearest Neighbors) là hai trong số các thuật toán phân loại cho bộ dữ liệu số phổ biến trong lĩnh vực học máy. Trong đó, mỗi mô hình có một số các ưu và nhược điểm như sau:

1) Mô hình phân loại SVM (Support Vector Machine): SVM là một thuật toán phân loại mạnh mẽ được sử dụng để tìm ra ranh giới quyết định tối ưu giữa các lớp dữ liệu. Mục tiêu của SVM là tạo ra một siêu phẳng (hyperplane) trong không gian n chiều sao cho khoảng cách từ các điểm dữ liệu gần nhất đến siêu phẳng đó là lớn nhất. SVM có khả năng xử lý cả dữ liệu tuyến tính và phi tuyến tính thông qua các hàm kernel như hàm đa thức, hàm Radial Basis Function (RBF), hoặc hàm sigmoid.

- Ưu điểm: Hiệu quả trong việc xử lý dữ liệu có số chiều lớn, linh hoạt với các hàm kernel khác nhau, có khả năng xử lý cả dữ liệu phi tuyến tính.

- Nhược điểm: Đòi hỏi bộ nhớ lớn và thời gian tính toán cao khi xử lý các tập dữ liệu lớn.

2) Mô hình phân loại kNN (k-Nearest Neighbors): kNN là một thuật toán phân loại đơn giản dựa trên việc so sánh các điểm dữ liệu với các điểm lân cận. Ý tưởng cơ bản của kNN là gán nhãn cho một điểm dữ liệu mới bằng cách xác định nhãn của các điểm lân cận gần nhất với nó. Độ đo thường được sử dụng trong kNN là khoảng cách Euclide hoặc khoảng cách Mahalanobis. Tham số k trong kNN là số lượng điểm lân cận được sử dụng để quyết định nhãn của điểm dữ liệu mới.

- Ưu điểm: Dễ triển khai, không cần giả định về phân phối của dữ liệu, linh hoạt với các loại hàm độ đo khác nhau.

- Nhược điểm: Đòi hỏi lưu trữ toàn bộ tập dữ liệu, độ phức tạp tính toán cao khi xử lý các tập dữ liệu lớn.

1.6.3.2. Độ đo đánh giá

Trong học máy, ma trận gây lẫn nhị phân (binary confusion matrix) thường được sử dụng để đánh giá hiệu suất của một mô hình phân loại trên một tập dữ liệu kiểm tra. Ma trận này có dạng như Bảng 1.9. Trong đó:

- TP (True Positive): Số lượng các mẫu thuộc lớp Positive (dương) được mô hình dự đoán đúng.

- TN (True Negative): Số lượng các mẫu thuộc lớp Negative (âm) được mô hình dự đoán đúng.

- FP (False Positive): Số lượng các mẫu thuộc lớp Negative nhưng được mô hình dự đoán là Positive.

- FN (False Negative): Số lượng các mẫu thuộc lớp Positive nhưng được mô hình dự đoán là Negative.

Bảng 1.9: *Ma trận lầm lẫn nhị phân*

Lớp thực tế	Lớp dự đoán	
	Dương (P)	Âm (N)
Đúng (T)	TP	TN
Sai (F)	FP	FN

Dựa trên ma trận gây lẫn nhị phân này, ta có thể tính toán nhiều độ đo đánh giá hiệu suất của mô hình như độ chính xác (accuracy), độ nhạy (recall), độ chính xác của lớp dương (precision), và F1-score. Ma trận gây lẫn cung cấp cái nhìn tổng quan về khả năng dự đoán của mô hình trên cả hai lớp và giúp xác định các lỗi dự đoán mô hình có thể gặp phải. Sau đây là một số độ đo thường được sử dụng để đánh giá các mô hình phân lớp.

1. Độ đo - Accuracy:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}. \tag{1.15}$$

2. Độ đo - Error:

$$Error = \frac{FP+FN}{TP+TN+FP+FN} \tag{1.16}$$

3. Độ đo - Precision:

$$Precision = \frac{TP}{TP+FP} \tag{1.17}$$

4. Độ đo - Recall:

$$Recall = \frac{TP}{TP+FN} \tag{1.18}$$

5. Độ đo - F_β:

$$F_\beta = \frac{\left(1+\beta^2\right) \times \text{ precision } \times \text{ recall}}{\beta^2 \times \text{ precision } + \text{ recall}} \tag{1.19}$$

1.6.3.3. Phương pháp đánh giá

Phương pháp đánh giá chéo k-fold là một kỹ thuật quan trọng dùng để đánh giá mô hình trong lĩnh vực học máy, thường được sử dụng để đánh giá khả năng của các mô hình phân lớp và dự đoán. Các thức hoạt động của phương pháp này bao gồm các bước chính như sau

- Chia tập dữ liệu: Tập dữ liệu ban đầu được chia ngẫu nhiên thành k phần (folds) có kích thước bằng nhau.

- Lặp lại quá trình đánh giá: Quá trình đánh giá được lặp lại k lần. Trong mỗi lần lặp, một phần trong số k phần được chọn làm tập kiểm thử, còn 9 phần còn lại được sử dụng làm tập huấn luyện.

- Huấn luyện mô hình: Mô hình được huấn luyện trên tập huấn luyện.

- Đánh giá mô hình: Mô hình được đánh giá trên tập kiểm thử để đo lường khả năng phân lớp.

- Ghi nhận kết quả: Kết quả đánh giá, chẳng hạn như độ chính xác được ghi lại.

- Tính toán độ chính xác trung bình: Sau khi hoàn thành k lần lặp, độ chính xác trung bình của mô hình được tính toán bằng cách lấy trung bình cộng của các kết quả đánh giá từ mỗi lần lặp.

Ưu điểm: Phương pháp này giúp đánh giá hiệu suất của mô hình một cách khách quan và chính xác. Đảm bảo rằng mô hình được đánh giá trên nhiều phần dữ liệu khác nhau, giúp giảm thiểu hiện tượng overfitting và underfitting.

Nhược điểm: Tăng thời gian đánh giá so với một số phương pháp đánh giá khác, đặc biệt khi mô hình phức tạp và dữ liệu lớn. Không phù hợp cho các tập dữ liệu có cấu trúc đặc biệt, chẳng hạn như dữ liệu chuỗi thời gian. Tóm lại, phương pháp đánh giá chéo 10-fold là một trong những kỹ thuật phổ biến và đáng tin cậy để đánh giá hiệu suất của mô hình trong học máy.

1.7. Kết luận Chương 1

Chương 1 đã giới thiệu khái quát về bài toán chọn lọc đặc trưng, ý nghĩa và vai trò của chọn lọc đặc trưng trong các lĩnh vực học máy và khai thác dữ liệu. Nghiên cứu tổng quát các nghiên cứu liên quan về các nhóm phương pháp chọn lọc đặc trưng cải thiện nhiễu và nhóm phương pháp chọn lọc đặc trưng gia tăng. Trên cơ sở phân tích một số các nghiên cứu liên quan gần đây cho thấy sự kém hiệu quả về thời gian chọn lọc đặc trưng bắt nguồn từ một số nguyên nhân sau đây:

- Đối với phương pháp chọn lọc đặc trưng cải thiện độ chính xác phân lớp: có hai phương pháp tiếp cận chính là sử dụng mô hình VPFRS và mô hình IFRS. Phương pháp chọn lọc đặc trưng cải thiện độ chính xác phân lớp theo tiếp cận VPFRS dựa trên sự điều chỉnh độ chính xác của các phần tử trong tập xấp xỉ dưới. Tuy nhiên công thức tính tập xấp xỉ dưới còn chưa hiệu quả do phải xét toàn bộ các đối tượng trong

bảng quyết định cho mỗi tập xấp xỉ dưới. Phương pháp chọn lọc đặc trưng cải thiện độ chính xác phân lớp theo tiếp cận IFRS dựa trên nền tảng của lý thuyết tập mờ trực cảm. Tuy nhiên không gian xấp xỉ mờ trực cảm có độ phức tạp gấp đôi so với không gian xấp xỉ mờ. Hơn nữa, để xây dựng các công thức tính toán độ thành viên và không thành viên độc lập nhau, thỏa mãn điều kiện là một số mờ trực cảm là không dễ dàng. Đặc biệt là xây dựng công thức mô tả chính xác về độ tương tự và không tương tự, phù hợp cho mọi bảng dữ liệu là rất khó khăn và tốn kém. Do đó, luận án định hướng Chương 2 nghiên cứu mở rộng mô hình VPFRS, ứng dụng chọn lọc đặc trưng cải thiện độ chính xác phân lớp, hiệu quả về thời gian tính toán.

- Đối với phương pháp chọn lọc đặc trưng gia tăng khi bảng quyết định thay đổi về đối tượng: có hai phương pháp tiếp cận chính là sử dụng độ đo khoảng cách tri thức và độ đo hạt thông tin tri thức để xây dựng các công thức tính toán gia tăng, cập nhật tập rút gọn cho các trường hợp bổ sung và loại bỏ tập đặc trưng. Thông qua các kết quả phân tích các nghiên cứu gần đây cho thấy độ đo hạt thông tin tri thức cho phương pháp xây dựng công thức tính toán gia tăng đơn giản hơn độ đo khoảng cách. Tuy nhiên, độ đo hạt thông tin tri thức chưa được mở rộng trên không gian xấp xỉ mờ. Do đó, luận án định hướng Chương 3 nghiên cứu mở rộng độ đo hạt thông tin tri thức trên không gian xấp xỉ mờ, ứng dụng xây dựng các công thức tính toán gia tăng, cập nhật tập rút gọn khi tập đối tượng thay đổi.

CHƯƠNG 2. RÚT GỌN THUỘC TÍNH THEO TIẾP CẬN MỞ RỘNG TẬP THÔ MỜ ĐỘ CHÍNH XÁC THAY ĐỔI TRÊN KHÔNG GIAN XẤP XỈ MỜ

Chương 2 của luận án tập trung vào bài toán chọn lọc thuộc tính cải thiện độ chính xác phân lớp cho bảng quyết định số có chứa nhiễu. Xuất phát từ hạn chế về thời gian tính toán của các phương pháp dựa trên mô hình tập thô mờ điều chỉnh (VPFRS) và không gian xấp xỉ mờ trực cảm (IFRS), luận án đề xuất cải tiến mô hình VPFRS nhằm tối ưu hóa thời gian tính toán các tập xấp xỉ.

Điểm nổi bật của chương là việc đề xuất mô hình tập thô mờ độ chính xác mở rộng (VPOFRS) thông qua việc cải tiến các phép toán về cách tính độ thuộc. Bên cạnh đó, luận án cũng xây dựng độ đo độ phụ thuộc mới dựa trên mô hình VPOFRS và đề xuất thuật toán rút gọn thuộc tính theo tiếp cận VPOFRS, đây là thuật toán được thiết kế theo phương pháp filter với độ phức tạp đa thức. Tuy nhiên với các cải tiến của mô hình VPOFRS sẽ hứa hẹn giảm đáng kể thời gian tính toán so với các phương pháp trước đó.

Kết quả thực nghiệm cho thấy, thuật toán VPOFRS_ARD có thời gian thực hiện nhanh hơn so với VPFRS và IFRS, đồng thời cho tập rút gọn có kích thước nhỏ hơn và độ chính xác phân lớp cao hơn trên một số bộ dữ liệu nhiễu, đặc biệt là trên bộ dữ liệu UFDC. Điều này khẳng định tính hiệu quả của phương pháp đề xuất trong việc giảm thiểu ảnh hưởng của nhiễu và cải thiện hiệu suất tính toán.

Các kết quả nghiên cứu được công bố trong các công trình [CT2, CT3]. Trước khi đi vào nội dung chính, các kiến thức cơ sở được nhắc lại làm nền tảng để xây dựng các đề xuất trong chương nghiên cứu.

2.1. Các kiến thức cơ sở

Phần này nhắc lại một số kiến thức quan trọng về mô hình tập thô VPRS do Ziarko đề xuất vào năm 1993, ứng dụng rút gọn thuộc tính cải thiện độ chính xác phân lớp cho bảng quyết định phân loại không đầy đủ. Mô hình tập thô mờ VPFRS được Zhao giới thiệu vào năm 2009 [48], là mô hình mở rộng của mô hình VPRS trên không gian xấp xỉ mờ. Mô hình VPFRS được ứng dụng hiệu quả trong rút gọn thuộc tính trên các bộ dữ liệu số có chứa nhiễu.

2.1.1. Mô hình tập thô điều chỉnh

Định nghĩa 2.1 (Tỉ lệ thuộc)**.** [28]Cho U là tập không rỗng các đối tượng, với mọi $X,Y \subseteq U$, X được gọi là tập con của Y nếu với mọi $x \in X$ ta có $x \in Y$. Độ thuộc của X vào Y được xác định bởi công thức sau đây:

$$c(X,Y) = \begin{cases} 1 - \frac{|X \cap Y|}{|X|} \Leftrightarrow |X| > 0 \\ 0 \Leftrightarrow |X| = 0 \end{cases} \tag{2.1}$$

Trong đó $|.|$ là kí hiệu lực lượng của một tập.

Mệnh đề 2.1 (Độ bao thuộc)**.** *[28] Dựa trên công thức (2.1), độ bao thuộc của Y với X được đánh giá theo công thức như sau*

$$X \subseteq Y \Leftrightarrow c(X,Y) = 0 \tag{2.2}$$

Định nghĩa 2.2 (β thuộc)**.** [28] Dựa trên công thức (2.2), độ thuộc của X với Y tại ngưỡng β được xác định bởi công thức sau:

$$X \overset{\beta}{\subseteq} Y \Leftrightarrow c(X,Y) \leqslant \beta \tag{2.3}$$

Định nghĩa 2.3 (Mô hình tập thô điều chỉnh)**.** [28] Dựa trên công thức (2.3), Mô hình tập thô điều chỉnh được định nghĩa lại như sau:

Với mọi $X \subseteq U$, β - xấp xỉ dưới của X được xác định bởi:

$$\underline{R}_{\beta}X = \bigcup \{E \in U/R : c(E,X) \leqslant \beta\} \tag{2.4}$$

Với mọi $X \subseteq U$, β - xấp xỉ trên của X được xác định bởi:

$$\bar{R}_{\beta}X = \bigcup \{E \in U/R : c(E,X) < 1-\beta\} \tag{2.5}$$

Trong đó U/R là phân hoạch của U theo quan hệ tương đương R.

2.1.2. Mô hình tập thô mờ điều chỉnh

Định nghĩa 2.4 (Quan hệ tương đương mờ)**.** [28] Cho U là tập không rỗng các đối tượng và $\mathscr{R}$ là một quan hệ trên U, khi đó $\mathscr{R}$ được gọi là quan hệ tương đương mờ nếu các điều kiện sau đây được thỏa mãn. Với mọi $x, y, z \in U$.

(1) Tính phản xạ: $\mathscr{R}(x,x) = 1$,

(2) Tính đối xứng $\mathscr{R}(x,y) = \mathscr{R}(y,x)$,

(3) Tính bắc cầu $\mathscr{R}(x,y) \geqslant \min(\mathscr{R}(x,z), \mathscr{R}(z,y))$.

Dựa trên toán tử t-chuẩn (min) và toán tử kéo théo $\mathscr{I}$ của tập mờ. Mô hình tập thô mờ được định nghĩa dựa trên các phép toán xấp xỉ sau đây.

Định nghĩa 2.5 (Tập thô mờ)**.** [1]Cho U là tập không rỗng các đối tượng, với mọi $x \in U, A \subseteq U$,

Độ thuộc của x với tập xấp xỉ dưới của A theo quan hệ $\mathscr{R}$ trên U được xác định bởi:

$$\underline{\mathscr{R}}A(x) = \inf_{y \in U} \mathscr{I}(\mathscr{R}(x,y), A(y)) \tag{2.6}$$

Độ thuộc của x với tập xấp xỉ trên của A theo quan hệ $\mathscr{R}$ trên U được xác định bởi:

$$\overline{\mathscr{R}}A(x) = \sup_{y \in U} \min(\mathscr{R}(x,y), A(y)) \tag{2.7}$$

Dựa trên khái niệm điều chỉnh trong mô hình VPRS, Zhao đề xuất mô hình tập thô mờ điều chỉnh như sau:

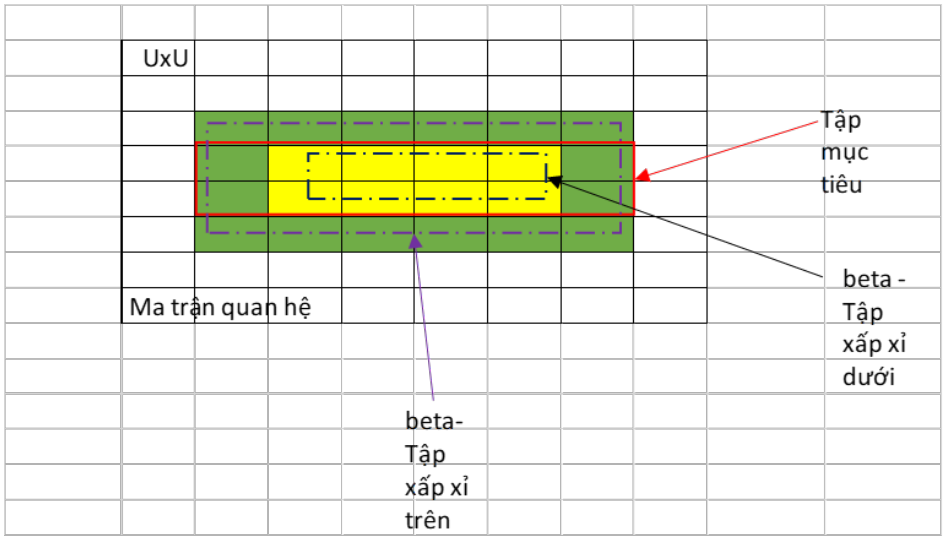
Định nghĩa 2.6 (Tập thô mờ điều chỉnh)**.** [1]Cho U là tập không rỗng các đối tượng, với mọi $x, y \in U$, $A \subseteq U$,

β-thuộc của x với tập xấp xỉ dưới của A theo quan hệ $\mathscr{R}$ trên U được xác định như sau:

$$\underline{\mathscr{R}}_{\beta}A(x) = \inf_{A(y)\leq\beta} \mathscr{I}(\mathscr{R}(x,y),\beta) \wedge \inf_{A(y)>\beta} \mathscr{I}(\mathscr{R}(x,y),A(y)) \tag{2.8}$$

β-thuộc của x với tập xấp xỉ dưới của A theo quan hệ $\mathscr{R}$ trên U được xác định như sau:

$$\overline{\mathscr{R}_{\beta}}A(x) = \sup_{A(y)\geq 1-\beta} \min(\mathscr{R}(x,y), 1-\beta) \vee \sup_{A(y)<1-\beta} \min(\mathscr{R}(x,y),A(y)) \tag{2.9}$$



Hình 2.1: *Mô hình lý thuyết tập thô mờ điều chỉnh*

Một cách hình thức, mô hình lý thuyết tập thô mờ điều chỉnh có thể được biểu diễn theo Hình 2.1

2.2. Đề xuất phương pháp cải tiến mô hình tập thô mờ điều chỉnh

Theo công thức (2.8) và công thức (2.9) ta có thể thấy phương pháp tính toán các tập xấp xỉ này là chưa tối ưu do phải xét toàn bộ $y \in U$. Hơn nữa, các công thức này

phải tính toán độ thuộc của x dựa trên hai khoảng của β và $1-\beta$. Sau đây là phần trình bày phương pháp mở rộng mô hình VPFRS dựa theo các phép toán xấp xỉ mờ ngưỡng β được định nghĩa lại. Trước tiên là phần định nghĩa về không gian xấp xỉ mờ.

2.2.1. Không gian xấp xỉ mờ đề xuất

Định nghĩa 2.7 (Quan hệ tương đương mờ)**.** Cho U là tập không rỗng các đối tượng, với mọi $x, y \in U$, độ tương tự của hai x và y được xác định như sau:

$$\mathscr{R}(x,y) = 1 - |x-y| \tag{2.10}$$

Nhận thấy $\mathscr{R}$ là một quan hệ tương đương mờ trên U. Khi đó $[x]_{\mathscr{R}}$ là một lớp tương đương mờ của $x \in U$. Cặp $\mathscr{R}, U$ được gọi là không gian xấp xỉ mờ của $\mathscr{R}$ trên U.

Định nghĩa 2.8 (Lớp tương đương mờ)**.** Cho U là tập không rỗng các đối tượng và quan hệ tương đương mờ $\mathscr{R}$ xác định trên U. Khi đó, lớp tương đương của $x \in U$ theo quan hệ $\mathscr{R}$ được xác định như sau:

$$[x]_{\mathscr{R}} = \{y, \mathscr{R}(x,y) : y \in U\} \tag{2.11}$$

Nhận thấy $[x]_{\mathscr{R}}$ là một tập mờ trên U.

Định nghĩa 2.9 (Phân hoạch mờ)**.** Cho bảng quyết định $NDT = (U, C, D)$, và $\mathscr{R}$ là một quan hệ tương đương mờ trên U. Khi đó, phân hoạch mờ của $\mathscr{R}$ trên U được xác định như sau:

$$U/\mathscr{R} = \{[x]_{\mathscr{R}} : x \in U\} \tag{2.12}$$

Định nghĩa 2.10 (Phân hoạch mờ của một thuộc tính)**.** Cho bảng quyết định $NDT = (U, C, D)$, và $\mathscr{R}$ là một quan hệ tương đương mờ trên U. Khi đó, phân hoạch mờ của U theo thuộc tính $a \in C$ trên $\mathscr{R}$ được kí hiệu bởi $U/\mathscr{R}_a$. Trong đó $U/\mathscr{R}_a$ được xác định theo công thức (2.12)

Định nghĩa 2.11 (Phân hoạch mờ của tập thuộc tính)**.** Cho bảng quyết định $NDT = (U, C, D)$, và $\mathscr{R}$ là một quan hệ tương đương mờ trên U. Khi đó, phân hoạch mờ của U theo tập thuộc tính $P \subseteq C$ trên $\mathscr{R}$ được kí hiệu bởi $\mathscr{P}$, với $\mathscr{P}$ được xác định như sau:

$$\mathscr{P} = \bigcap_{a \in P} U/R_a \tag{2.13}$$

Mệnh đề 2.2 (Tính đơn điệu của phân hoạch mờ)**.** *Cho bảng quyết định $NDT = (U, C, D)$ và $\mathscr{P}, \mathscr{Q}$ là hai phân hoạch mờ trên U tương ứng với $P, Q \subseteq C$. Khi đó, nếu $\mathscr{P} \subseteq \mathscr{Q}$ thì $Q \subseteq P$.*

Chứng minh. Giả sử $P = Q \cup \{b\} : b \in C$ Theo công thức (2.13) và toán tử t-chuẩn (min) của tập mờ ta có $U/R_b \cap \bigcap_{a \in Q} U/R_a \leq \bigcap_{a \in Q} U/R_a$, nghĩa là $\mathscr{P} \subseteq \mathscr{Q}$. Ta có điều phải chứng minh. □

Định nghĩa 2.12. Cho bảng quyết định $NDT = (U, C, D)$ và $\mathscr{P}$ là hai phân hoạch mờ của U với $P \subseteq C$. Khi đó $\mathscr{P}$ được gọi là phân hoạch mịn nhất nếu mọi $x \in U$, $[x]_{\mathscr{P}} = \{x\}$.

Định nghĩa 2.13. Cho bảng quyết định $NDT = (U, C, D)$ và $\mathscr{Q}$ là hai phân hoạch mờ của U với $Q \subseteq C$. Khi đó $\mathscr{Q}$ được gọi là phân hoạch thô nhất nếu mọi $x \in U$, $[x]_{\mathscr{Q}} = U$.

2.2.2. Mô hình tập thô mờ điều chỉnh đề xuất

Dựa trên các vấn đề còn tồn tại của các công thức tính tập xấp xỉ (2.8) và (2.9), phần này sẽ trình bày phương pháp cải tiến các phép toán về cách tính độ thuộc của $x \in U$ và cách xác định giới hạn của $y \in U$

Định nghĩa 2.14 (Cải tiến độ thuộc của $x \in U$)**.** Cho U là tập không rỗng các đối tượng, với mọi $x, y \in U$, $A \subseteq U$. Khi đó:

β-thuộc của x với tập xấp xỉ dưới của A theo quan hệ $\mathscr{R}$ trên U được xác định như

sau:

$$\underline{\mathscr{R}_\beta}A(x) = \min\{\inf_{y\in U} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} \tag{2.14}$$

β-thuộc của x với tập xấp xỉ trên của A theo quan hệ $\mathscr{R}$ trên U được xác định như sau:

$$\overline{\mathscr{R}_\beta}A(x) = \max\{\sup_{y\in U}\min(\mathscr{R}(x,y),A(y)),\beta\} \tag{2.15}$$

Mệnh đề 2.3 (Cải tiến giới hạn của $y \in U$)**.** *Cho bảng quyết định $NDT = (U,C,D)$. Với mọi $y \in U$ và $A \subseteq U$ ta có:*

$$min\{\inf_{y\in U} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} = min\{\inf_{y\in A} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} \tag{2.16}$$

$$max\{\sup_{y\in U} min(\mathscr{R}(x,y),A(y)),\beta\} = max\{\sup_{y\in A} min(\mathscr{R}(x,y),A(y)),\beta\} \tag{2.17}$$

Chứng minh. Dựa theo công thức (2.14) và (2.15) ta có:

(1) Nếu $y \in A$ thì $A(y) = 1$.

Do đó ta có $\min\{\inf_{y\in U} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} = \min\{\inf_{y\in A} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} = \min\{1,\beta\}$.

Nếu $y \notin A$ thì $A(y) = 0$.

Khi đó ta có $\min\{\inf_{y\in U} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} = \min\{\inf_{y\in A} \mathscr{I}(\mathscr{R}(x,y),A(y)),\beta\} = \min\{0,\beta\}$.

(2) Nếu $y \in A$ thì $A(y) = 1$.

Do đó ta có $\max\{\sup_{y\in U} \min(\mathscr{R}(x,y),A(y)),\beta\} = \max\{\sup_{y\in A} \min(\mathscr{R}(x,y),A(y)),\beta\} = \max\{\mathscr{R}(x,y),\beta\}$.

Nếu $y \notin A$ thì $A(y) = 0$.

Khi đó ta có $\max\{\sup_{y\in U} \min(\mathscr{R}(x,y),A(y)),\beta\} = \max\{\sup_{y\in A} \min(\mathscr{R}(x,y),A(y)),\beta\} = \beta$.

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

Dựa trên mệnh đề 3.2, mô hình VPFRS được định nghĩa lại như sau:

Định nghĩa 2.15 (Mô hình VPOFRS)**.** Cho U là tập không rỗng các đối tượng, với mọi $x, y \in U$, $A \subseteq U$. Khi đó

β-thuộc của x với tập xấp xỉ dưới của A theo quan hệ $\mathscr{R}$ trên U được xác định như sau:

$$\underline{\mathscr{R}}_{\beta}A(x) = \min\{\inf_{y\in A} \mathscr{I}(\mathscr{R}(x,y), A(y)), \beta\} \tag{2.18}$$

Khi đó, β-xấp xỉ dưới của A theo $\mathscr{R}$ là một tập mờ được xác định bởi

$$\underline{\mathscr{R}}_{\beta}A = \bigcup_{x\in A} \{\underline{\mathscr{R}}_{\beta}A(x)\} \tag{2.19}$$

β-thuộc của x với tập xấp xỉ trên của A theo quan hệ $\mathscr{R}$ trên U được xác định như sau:

$$\overline{\mathscr{R}_{\beta}}A(x) = \max\{\sup_{y\in A} \min(\mathscr{R}(x,y), A(y)), \beta\} \tag{2.20}$$

Khi đó, β-xấp xỉ trên của A theo $\mathscr{R}$ là một tập mờ được xác định bởi

$$\overline{\mathscr{R}_{\beta}}A = \bigcup_{x\in A} \{\overline{\mathscr{R}_{\beta}}A(x)\} \tag{2.21}$$

Cặp $(\underline{\mathscr{R}}_{\beta}A(x), \overline{\mathscr{R}_{\beta}}A(x))$ được gọi là mô hình VPOFRS.

Rõ ràng $\underline{\mathscr{R}}_{\beta}A$ luôn luôn nhỏ hơn hoặc bằng $\overline{\mathscr{R}_{\beta}}A$. Nếu $\underline{\mathscr{R}}_{\beta}A = \overline{\mathscr{R}_{\beta}}A$, ta gọi A là tập β - chính xác.

Dựa theo mối quan hệ của $\underline{\mathscr{R}}_{\beta}A(x)$ và $\overline{\mathscr{R}_{\beta}}A(x)$, tập U sẽ được phân chia thành các miền theo A như sau:

- β - miền dưới (miền chắc chắn) của A được xác định bởi

$$L_{\beta}A = \bigcup_{x\in A} \{\underline{\mathscr{R}}_{\beta}A(x)\} \tag{2.22}$$

Khi đó, β - miền dưới của A là một tập mờ trong $F(U)$ chắc chắn thuộc A với tỉ lệ

lỗi β hay:

$$L_\beta A = A \cap L_\beta A \tag{2.23}$$

- β - miền trên (miền không chắc chắn) của A được xác định bởi

$$H_\beta A = \bigcup_{x \in A} \{\overline{\mathscr{R}_\beta} A(x)\} \tag{2.24}$$

Khi đó, β - miền trên của A là một tập mờ trong $F(U)$ không chắc chắn thuộc A với tỉ lệ lỗi β hay:

$$A = A \cap H_\beta A \tag{2.25}$$

- β - miền biên (miền trung gian) của A được xác định bởi:

$$B_\beta A = H_\beta A - L_\beta A \tag{2.26}$$

Khi đó, β - miền biên của A là một tập mờ trong $F(U)$ vừa thuộc $H_\beta A$, vừa thuộc $L_\beta A$ với tỉ lệ lỗi β hay $B_\beta A \cap H_\beta A \neq \emptyset$ và $B_\beta A \cap H_\beta A \neq \emptyset$.

- β - miền âm (miền phủ định) của A được xác định bởi:

$$N_\beta A = U - \overline{\mathscr{R}_\beta} A \tag{2.27}$$

β - miền âm của A là một tập mờ trong $F(U)$ hoàn toàn không thuộc A với tỉ lệ lỗi β hay:

$$A \cap N_\beta A = \emptyset \tag{2.28}$$

2.2.3. Độ đo độ phụ thuộc đề xuất

Dựa vào mô hình VPOFRS được mở rộng từ mô hình VPFRS, độ đo độ phụ thuộc được xây dựng ứng dụng cho phân tích dữ liệu của bảng quyết định. Trước tiên, khái niệm miền dương của bảng quyết định được định nghĩa.

Định nghĩa 2.16 (β - miền dương của bảng quyết định)**.** Cho bảng quyết định $NDT = (U,C,D)$ với $D = \{D_1, D_2, ...D_n\}$ và $\mathscr{C}$ là phân hoạch mờ của C trên U. Khi đó β - miền dương của bảng quyết định NDT được định nghĩa như sau:

$$POS_\beta(C,D) = \bigcup \{\underline{\mathscr{C}}_\beta X : X \in D\} \tag{2.29}$$

Định nghĩa 2.17 (Bảng quyết định nhất quán - β)**.** Cho bảng quyết định $NDT = (U,C,D)$, NDT được gọi là bảng quyết định nhất quán nếu:

$$POS_\beta(C,D) = U \tag{2.30}$$

Định nghĩa 2.18 (Độ nhất quán - β)**.** Cho bảng quyết định $NDT = (U,C,D)$, độ nhất quán β của bảng quyết định được xác định như sau

$$\gamma_\beta(C,D) = \frac{POS_\beta(C,D)}{|U|} \tag{2.31}$$

Nhận thấy bảng quyết định NDT được gọi là nhất quán - β nếu $\gamma_\beta(C,D) = 1$, khi đó D hoàn toàn phụ thuộc vào C. Độ đo này cũng thể hiện mức độ phụ thuộc của thuộc tính D trong bảng quyết định. Sau đây được gọi chung là độ đo độ phụ thuộc.

Mệnh đề 2.4 (Tính đơn điệu của độ phụ thuộc - β)**.** *Cho bảng quyết định* $NDT = (U,C,D)$ *và* $P,Q \subseteq C$*, khi đó* $\gamma_\beta(P,D) \leqslant \gamma_\beta(Q,D)$ *if* $P \subseteq Q$*.*

Chứng minh. Dựa theo mệnh đề 2.2 ta có: nếu $P \subseteq Q$, thì $\mathscr{Q} \subseteq \mathscr{P}$. Do đó, với mọi $x \in U$ và $A \subseteq U$, nếu $[x]_P$ thuộc A, thì $[x]_Q$ cũng thuộc A. Khi đó $\underline{\mathscr{P}}_\beta A(x) \leq \underline{\mathscr{Q}}_\beta A(x)$ do đó $POS_\beta(P,A) \subseteq POS_\beta(Q,A)$. Khi đó ta có $\frac{POS_\beta(P,D)}{|U|} \leq \frac{POS_\beta(Q,D)}{|U|}$ hay $\gamma_\beta(P,D) \leq \gamma_\beta(Q,D)$. Ta có điều phải chứng minh. □

2.3. Đề xuất phương pháp rút gọn thuộc tính cải thiện độ chính xác phân lớp

Dựa trên độ đo độ nhất quán của bảng quyết định được đề xuất trong phần trên của chương nghiên cứu. Phần này sẽ trình bày phương pháp rút gọn thuộc tính cải thiện

nhiễu theo theo tiếp cận VPOFRS. Trước tiên là phần trình bày về mô hình đề xuất.

2.3.1. Mô hình đề xuất

Dựa trên tính chất đơn điệu của độ đo độ nhất quán - β đã được đề xuất trong phần trên của chương nghiên cứu, phần này xây dựng độ đo đánh giá độ quan trọng của thuộc tính và định nghĩa tập rút gọn.

Định nghĩa 2.19 (Độ quan trọng thuộc tính ngưỡng β)**.** Cho bảng quyết định $NDT = (U, C, D)$ và tập thuộc tính $B \subseteq C$. Khi đó, độ quan trọng của thuộc tính $c \in \{C - B\}$ với B tại ngưỡng β được xác định bởi:

$$Sig_\beta(c, B) = \gamma_\beta(B \cup \{c\}, D) - \gamma_\beta(B, D) \tag{2.32}$$

Mệnh đề 2.5 (Tính chất khoảng cách)**.** *Cho bảng quyết định $NDT = (U, C, D)$ với tập thuộc tính $B \subseteq C$. Khi đó $Sig_\beta(c, B) \geq 0$ với mọi $c \in C - B$.*

Chứng minh. Dựa theo tính chất đơn điệu của mệnh đề 2.4 ta có $\gamma_\beta(B \cup \{c\}, D) \geq \gamma_\beta(B, D)$. Do đó $Sig_\beta(c, B) \geq 0$ với mọi $c \in C - B$. Ta có điều phải chứng minh. □

Như vậy dựa theo mệnh đề 2.5 ta có $c \in C - B$ được gọi là thuộc tính không quan trọng với B nếu $Sig_\beta(c, B) = 0$, hay khoảng cách phụ thuộc trước và sau khi bổ sung thuộc tính c vào B không thay đổi. Trên cơ sở đó, tập rút gọn theo tiếp cận VPOFRS được định nghĩa như sau.

Định nghĩa 2.20. Cho bảng quyết định $NDT = (U, C, D)$, khi đó $B \subseteq C$ được gọi là β-tập rút gọn nếu các điều kiện sau đây được thỏa mãn:

(1) Điều kiện cần: $\gamma_\beta(D, B) = \gamma_\beta(D, C)$

(2) Điều kện đủ: Với mọi $b \in B, \gamma_\beta(D, B - \{b\}) \neq \gamma_\beta(D, B)$

Định nghĩa 2.20 cho thấy điều kiện thứ nhất cần thỏa mãn nhằm đảm bảo tính bảo toàn độ nhất quán của tập thuộc tính rút bọn B so với tập thuộc tính gốc C, điều kiện thứ hai cần thỏa mãn để đảm bảo tính tối thiểu của tập rút gọn B.

2.3.2. Thuật toán đề xuất

Dựa trên các định nghĩa về độ quan trọng của thuộc tính và tập rút gọn mức β được xây dựng trong phần trên của chương nghiên cứu. Phần này trình bày thuật toán VPOFRS_ARD, rút gọn thuộc tính cải thiện độ chính xác phân lớp theo tiếp cận VPOFRS.

thuật toán VPOFRS_ARD được thiết kế theo phương pháp filter thuộc tính, với chiến lược tìm tập rút gọn được thực hiện qua các bước như sau. Với mỗi giá trị $\beta \in [0,1]$:

- Khởi tạo tập $B = \emptyset$ tại bước 1;

- Thực hiện cập nhật B tại các bước từ 3-11. Trong đó:

1) Xác định thuộc tính b trong $C - B$ có độ quan trọng lớn nhất với B theo định nghĩa 2.19, tại các bước từ 5-6.

2) bổ sung thuộc tính b vào B.

- Các bước từ 3-11 sẽ kết thúc khi điều kiện bảo toàn theo định nghĩa 2.20 tại bước số 2 được thỏa mãn.

- Kết thúc thuật toán ta thu được tập rút gọn B mức β.

Sau đây là nội dung chi tiết của thuật toán đề xuất:

Tiếp theo là phần đánh giá độ phức tạp của thuật toán VPOFRS_ARD. Cho bảng quyết đinh $NDT = (U, C, D)$. Kí hiệu $|U|$ là tập các đối tượng, $|C|$ là tập thuộc tính điều kiện, $|D|$ là số phân lớp và $|D_k|$ là số đối tượng của phân lớp thứ k trong $|D|$ của bảng quyết định. Dựa trên các công thức (2.29), (2.31) và (2.32) ta có. Độ phức tạp của các bước từ 5-10 là $\mathcal{O}(|D||D|_k|U||C|)$; Độ phức tạp của các bước từ 2-12 is $\mathcal{O}(|D||D_k||U||C|^2)$. Do đó độ phức tạp của thuật toán VPOFRS_ARD là $\mathcal{O}(|D||D_k||U||C|^2)$.

Độ phức tạp của thuật toán VPFRS, rút gọn thuộc tính theo tiếp cận VPFRS của Zhao đề xuất [48] có độ phức tạp là $\mathcal{O}(|D_k||U|^2|C|^2)$, do tập xấp xỉ dưới được tính dựa trên các phần tử trong U. Trong khi đó thuật toán đề tính tập xấp xỉ dưới chỉ

Thuật toán 2.1 Thuật toán VPOFRS_ARD rút gọn thuộc tính theo tiếp cận VPOFRS

Input Decision Table $DT = (U, C, D)$, $\beta \in [0, 1]$
Output The reduct B_β

```
1:  B = ∅;
2:  while γ_β(D,B) ≠ γ_β(D,C) do
3:     SIG = 0;
4:     b = ∅;
5:     for all c ∈ {C − B} do
6:        if Sig_β(c,B) > SIG then
7:           SIG = Sig_β(c,B);
8:           b = {c};
9:        end if
10:    end for
11:    B = B ∪ b;
12: end while
13: return B;
```

dựa trên các phần tử trong $|D_k|$. Trong thực tế $|D_k|$ luôn nhỏ hơn $|U|$. Đối với k phân lớp $|D_k| = \frac{|U|}{k}$, k càng tăng thì $|D_k|$ càng giảm và ngược lại. Do đó, về mặt lý thuyết, VPOFRS_ARD có thời gian thực hiện nhanh hơn thuật toán VPFRS.

2.3.3. Ví dụ số minh họa thuật toán

Để làm rõ tính đúng đắn của các công thức và thuật toán đề xuất. Phần này sẽ trình bày chi tiết các bước thực hiện của thuật toán qua từng công thức. Cụ thể như sau:

Đầu vào: Bảng quyết định $DT = (U, C, D)$, $\beta = 0.5$.

Đầu ra: Tập rút gọn B.

Bước 1: Khởi tạo $B = \emptyset$, $R_B = [1]_{|U|\times|U|}$. Trong đó, $[1]_{|U|\times|U|}$ là ma trận vuông kích thước $|U| \times |U|$ với mọi phần tử của ma trận có giá trị là 1.

Bước 2: Tính độ phụ thuộc của thuộc tính quyết định D vào tập thuộc tính B và tập thuộc tính C. Trước tiên ta áp dụng công thức (2.10) ta tính ma trận quan hệ của từng thuộc tính trong C như sau:

$$R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}; R_b = \begin{bmatrix} 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \end{bmatrix}$$

$$R_c = \begin{bmatrix} 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 1 & 0.2 & 0.4 & 0.4 & 1 \\ 0.8 & 0.2 & 1 & 0.8 & 0.8 & 0.2 \\ 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 1 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}; R_d = \begin{bmatrix} 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 0.8 & 1 & 0.6 & 0.8 & 0.8 & 0.6 \\ 0.8 & 0.6 & 1 & 0.8 & 0.8 & 0.2 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 0.6 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}$$

$$R_e = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.6 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 1 & 0.4 & 0.4 & 0.4 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \end{bmatrix}; R_f = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.6 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 1 & 0.4 & 0.4 & 0.4 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \end{bmatrix}$$

Áp dụng công thức (1.8) và phép toán T thứ nhất trong bảng 1.8 để xây dựng ma trận quan hệ của tập thuộc tính C như sau:

$$R_C = R_a \cap R_b \cap \ldots \cap R_f = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.6 & 0.2 & 1 & 0.4 & 0.4 & 0.2 \\ 0 & 0.2 & 0.4 & 1 & 0.8 & 0.4 \\ 0 & 0.2 & 0.4 & 0.8 & 1 & 0.4 \\ 0 & 0.2 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}$$

Thuộc tính quyết định D có hai phân lớp D_1 và D_2, trong đó $D_1 = \{x_1, x_3, x_6\}$ và $D_2 = \{x_2, x_4, x_5\}$. Nhận thấy hai phân lớp này có thể được biểu diễn tương ứng bởi hai

vector sau đây: $D_1 = \{x_1, x_3, x_6\} = [1,0,1,0,0,1]$, $D_2 = \{x_2, x_4, x_5\} = [0,1,0,1,1,0]$.

Áp dụng công thức (2.18), (2.29) và (2.31) ta tính được các độ đo độ phụ thuộc như sau:

$\gamma_\beta(B,D) = \frac{|[0,0,0,0,0,0]|}{|U|} = \frac{0}{6}$; $\gamma_\beta(C,D) = \frac{|[0.5,0.5,0.5,0.5,0.5,0.5]|}{|U|} = \frac{3}{6}$.

Bước 3: Vì $\gamma_\beta(B,D) = \frac{0.0}{6} \neq \gamma_\beta(C,D) = \frac{3}{6}$, thực hiện các bước trong vòng lặp ta có:

Áp dụng công thức (2.32):

- $Sig(a,B) = \gamma_\beta(B \cup a, D) - \gamma_\beta(B,D) = \frac{0.2}{6} - \frac{0.0}{6} = \frac{0.2}{6}$
- $Sig(b,B) = \gamma_\beta(B \cup b, D) - \gamma_\beta(B,D) = \frac{0.0}{6} - \frac{0.0}{6} = \frac{0.0}{6}$
- $Sig(c,B) = \gamma_\beta(B \cup c, D) - \gamma_\beta(B,D) = \frac{0.2}{6} - \frac{0.0}{6} = \frac{0.2}{6}$
- $Sig(d,B) = \gamma_\beta(B \cup d, D) - \gamma_\beta(B,D) = \frac{0.8}{6} - \frac{0.0}{6} = \frac{0.8}{6}$
- $Sig(e,B) = \gamma_\beta(B \cup e, D) - \gamma_\beta(B,D) = \frac{1.1}{6} - \frac{0.0}{6} = \frac{1.1}{6}$
- $Sig(f,B) = \gamma_\beta(B \cup f, D) - \gamma_\beta(B,D) = \frac{1.1}{6} - \frac{0.0}{6} = \frac{1.1}{6}$

Vì $Sig(e,B)$ là lớn nhất. Do đó, bổ sung thuộc tính e vào tập thuộc tính B ta có: $B = B \cup \{e\} = \{e\}$.

Vì $\gamma_\beta(B,D) = \frac{0.2}{6} \neq \gamma_\beta(C,D) = \frac{3}{6}$. Tiếp tục thực hiện vòng lặp ta có:

- $Sig(a,B) = \gamma_\beta(B \cup a, D) - \gamma_\beta(B,D) = \frac{1.3}{6} - \frac{0.2}{6} = \frac{1.1}{6}$
- $Sig(b,B) = \gamma_\beta(B \cup b, D) - \gamma_\beta(B,D) = \frac{1.5}{6} - \frac{0.2}{6} = \frac{1.3}{6}$
- $Sig(c,B) = \gamma_\beta(B \cup c, D) - \gamma_\beta(B,D) = \frac{2.4}{6} - \frac{0.2}{6} = \frac{2.2}{6}$
- $Sig(d,B) = \gamma_\beta(B \cup d, D) - \gamma_\beta(B,D) = \frac{2.7}{6} - \frac{0.2}{6} = \frac{2.5}{6}$
- $Sig(f,B) = \gamma_\beta(B \cup f, D) - \gamma_\beta(B,D) = \frac{1.1}{6} - \frac{0.2}{6} = \frac{0.9}{6}$

Vì $Sig(d,B)$ là lớn nhất. Do đó, bổ sung thuộc tính d vào tập thuộc tính B ta có: $B = B \cup \{d\} = \{e,d\}$.

Vì $\gamma_\beta(B,D) = \frac{2.5}{6} \neq \gamma_\beta(C,D) = \frac{3}{6}$. Tiếp tục thực hiện vòng lặp ta có:

- $Sig(a,B) = \gamma_\beta(B \cup a, D) - \gamma_\beta(B,D) = \frac{2.8}{6} - \frac{2.5}{6} = \frac{0.3}{6}$
- $Sig(b,B) = \gamma_\beta(B \cup b, D) - \gamma_\beta(B,D) = \frac{2.7}{6} - \frac{2.5}{6} = \frac{0.2}{6}$

- $Sig(c,B) = \gamma_\beta(B \cup c,D) - \gamma_\beta(B,D) = \frac{2.8}{6} - \frac{2.5}{6} = \frac{0.3}{6}$

- $Sig(f,B) = \gamma_\beta(B \cup f,D) - \gamma_\beta(B,D) = \frac{2.7}{6} - \frac{2.5}{6} = \frac{0.2}{6}$

Vì $Sig(a,B)$ là lớn nhất. Do đó, bổ sung thuộc tính a vào tập thuộc tính B ta có: $B = B \cup \{a\} = \{e,d,a\}$.

Vì $\gamma_\beta(B,D) = \frac{2.8}{6} \neq \gamma_\beta(C,D) = \frac{3}{6}$. Tiếp tục thực hiện vòng lặp ta có:

- $Sig(b,B) = \gamma_\beta(B \cup b,D) - \gamma_\beta(B,D) = \frac{2.8}{6} - \frac{2.8}{6} = \frac{0}{6}$

- $Sig(c,B) = \gamma_\beta(B \cup c,D) - \gamma_\beta(B,D) = \frac{3}{6} - \frac{2.8}{6} = \frac{0.2}{6}$

- $Sig(f,B) = \gamma_\beta(B \cup f,D) - \gamma_\beta(B,D) = \frac{2.8}{6} - \frac{2.8}{6} = \frac{0}{6}$

Vì $Sig(c,B)$ là lớn nhất. Do đó, bổ sung thuộc tính c vào tập thuộc tính B ta có: $B = B \cup \{c\} = \{e,d,a,c\}$.

Vì $\gamma_\beta(B,D) = \frac{3}{6} = \gamma_\beta(C,D) = \frac{3}{6}$, kết thúc vòng lặp While ta có tập rút gọn $B = \{e,d,a,c\}$.

2.4. Thực nghiệm và đánh giá

Để chứng minh thuật toán rút gọn thuộc tính đề xuất hiệu quả về thời gian rút gọn thuộc tính, phần này trình bày các kết quả thực nghiệm của thuật toán đề xuất và so sánh với thuật toán VPFRS [48] và thuật toán IFD [54].

2.4.1. Kịch bản và môi trường thực nghiệm

Để đánh giá tính hiệu quả về thời gian rút gọn thuộc tính của thuật toán đề xuất, đánh giá độ chính xác phân lớp và kích thước của tập rút gọn thu được, kịch bản thực nghiệm thuật toán đề xuất được thực hiện như sau

- Thứ nhất, khảo sát vùng giá trị β của thuật toán đề xuất cho ra tập rút gọn có kích thước và độ chính xác phân lớp tốt nhất. Trên cơ sở đó, sự biến động về độ chính xác phân lớp của tập rút gọn thu được tại mỗi giá trị β. Trên cơ sở đó, đánh giá được mối quan hệ đồng biến hay nghịch biến giữa độ chính xác phân lớp và kích thước của tập

rút gọn.

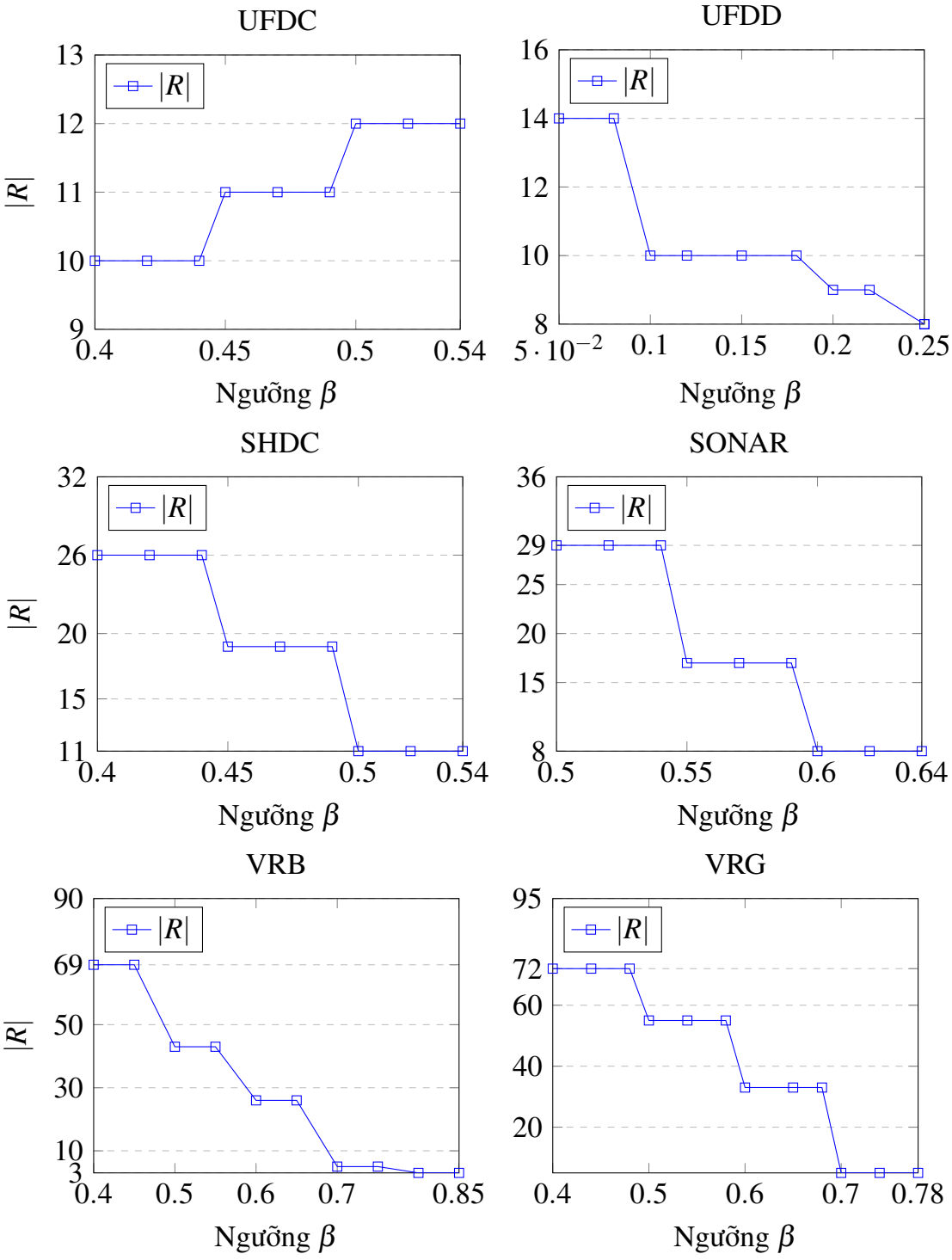
- Thứ hai, so sánh và đánh giá thuật toán đề xuất với các thuật toán rút gọn thuộc tính cải thiện độ chính xác gồm có thuật toán VPFRS của Zhao [48] đề xuất và thuật toán IFD của Nguyen [54] đề xuất. Ba tiêu chí dùng để so sánh gồm có: thời gian rút gọn thuộc tính, độ chính xác phân lớp và kích thước của tập rút gọn thu được từ các thuật toán.

Bảng 2.1: *Bảng mô tả các bộ dữ liệu có độ chính xác phân lớp thấp*

STT	Tập dữ liệu	Mô tả	$\|U\|$	$\|C\|$	$\|D\|$
1	UFDC	Ultrasonic flowmeter diagnostics (C)	181	43	4
2	UFDD	Ultrasonic flowmeter diagnostics (D)	181	43	4
3	SHDC	SPECTF Heart Data Set	267	44	2
4	SONAR	Connectionist Bench	208	60	2
5	VRB	Voice Rehabilitation(Binary)	126	310	2
6	VRG	Voice Rehabilitation(Gender)	126	310	2

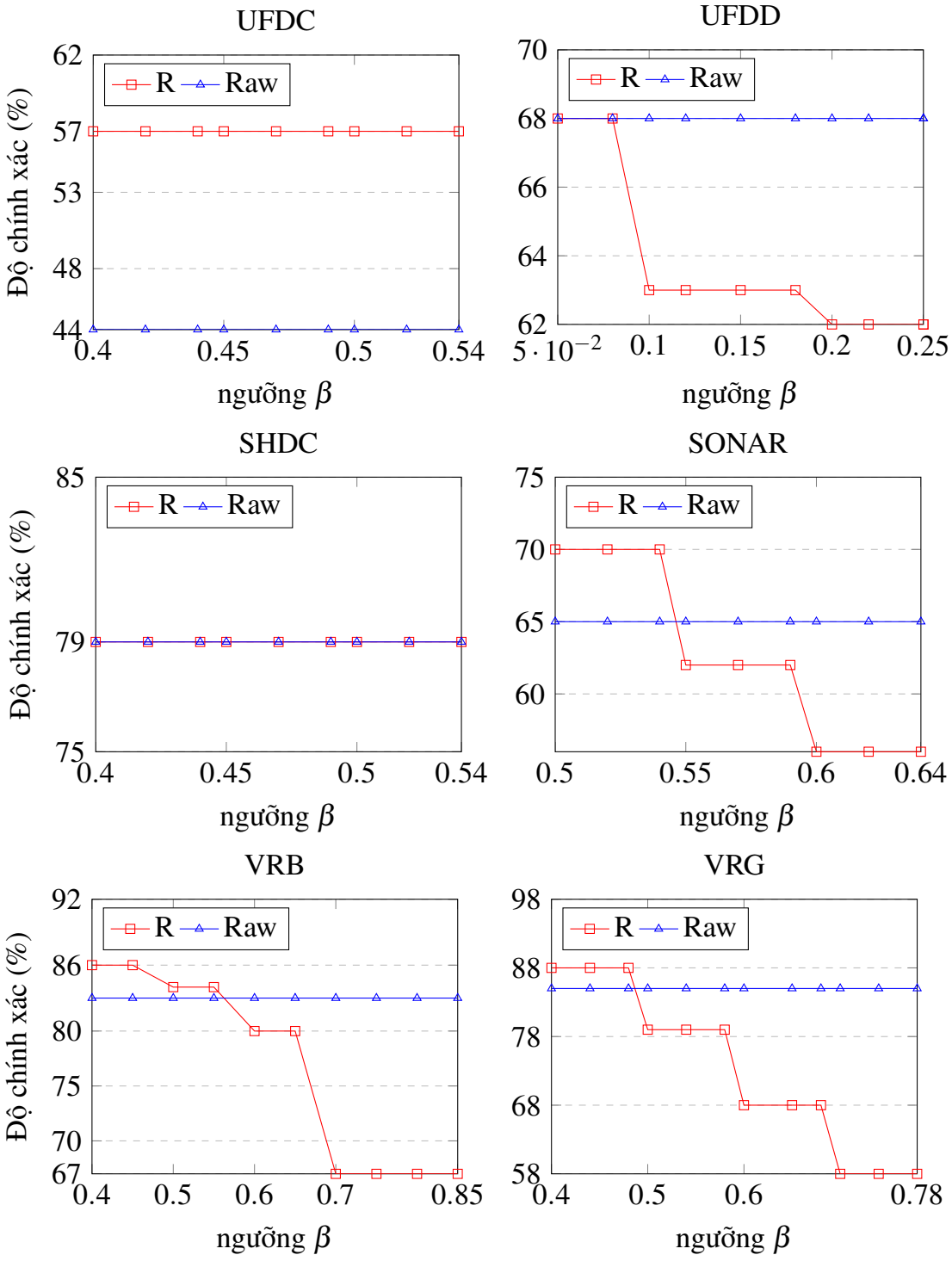
Các tập dữ liệu thực nghiệm được tải về từ kho dữ liệu chuẩn của UCI (UC-Irvine Machine Learning Repository) [93]. Tất cả các bộ dữ liệu này đều có độ chính xác phân lớp ban đầu thấp ($< 70\%$) được gọi là các bộ dữ liệu nhiễu. *Độ chính xác phân lớp của các tập dữ liệu này được đo lường bởi độ đo Accuracy kèm với phương pháp đánh giá chéo 10-folde trên hai bộ phân lớp k-NN và SVM*. Đây cũng là phương pháp đánh giá được sử dụng cho tập rút gọn thu được từ các thuật toán. Đặc biệt bộ dữ liệu UFDC có độ chính xác chỉ 44%. Chi tiết các bộ dữ liệu có thể tìm thấy tại Bảng 2.1, trong đó $|U|$ là số đối tượng, $|C|$ là số các thuộc tính điều kiện, và $|D|$ là số phân lớp của tập dữ liệu. Các thuộc tính điều kiện của các tập dữ liệu này đều có miền giá trị số liên tục. Trước khi thực hiện rút gọn thuộc tính với các thuật toán, dữ liệu được chuẩn hóa về đoạn $[0,1]$.

Tập rút gọn thu được từ các thuật toán được kiểm thử trên hai mô hình phân lớp dữ liệu số là SVM và k-NN với ($k = |D|$). Phương pháp đánh giá chéo 10-folde kết hợp với các độ đo (accuarcy, precision, recall) để đánh giá độ chính xác trung bình mô hình phân lớp xây dựng trên tập dữ liệu rút gọn. Ngôn ngữ lập trình Python được sử



Hình 2.2: *Mối quan hệ giữa kích thước tập rút gọn với ngưỡng* β

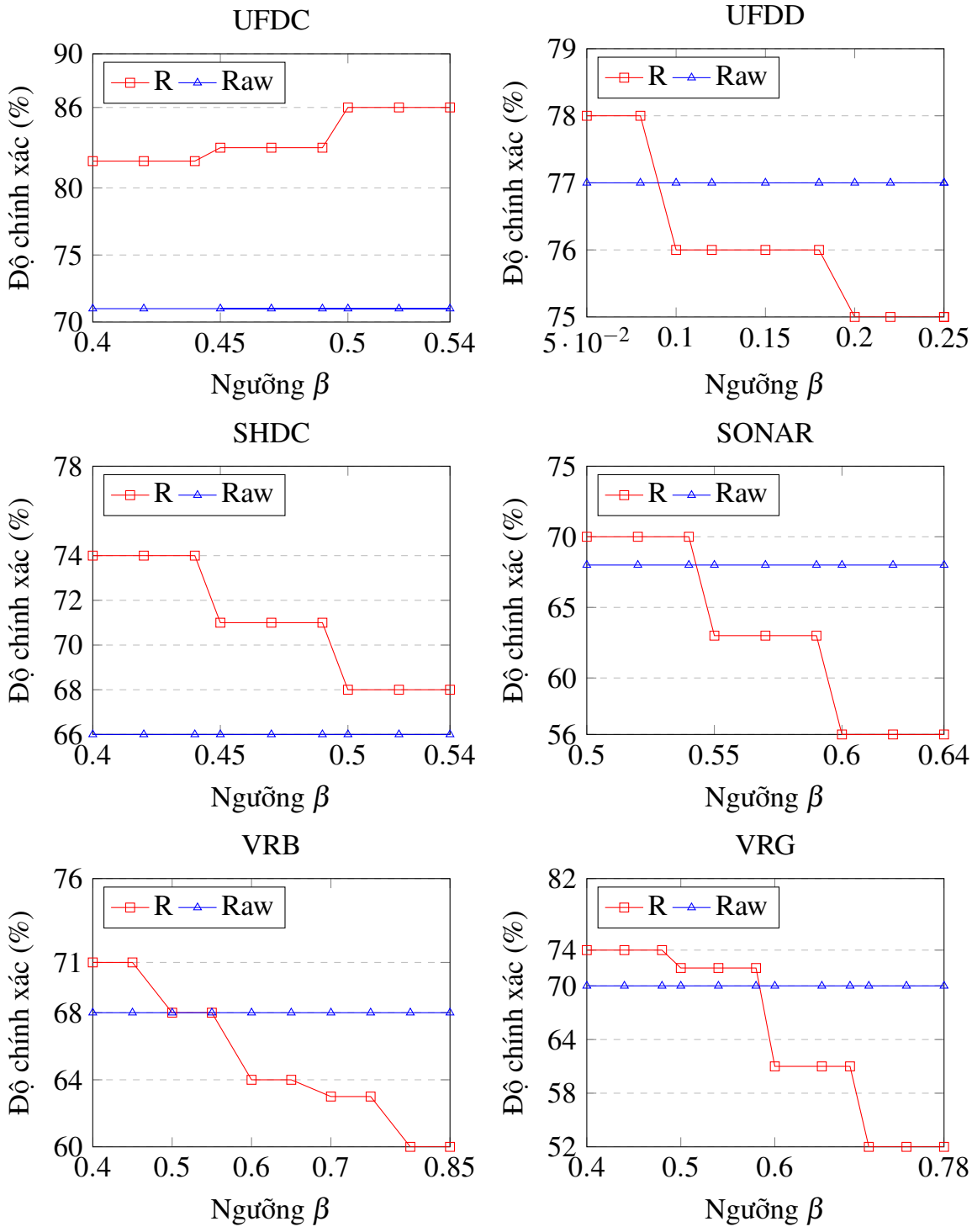
dụng để cài đặt các thuật toán trên nền tảng Windows 10 và phần cứng với bộ xử lý i5, 8GB RAM.



Hình 2.3: *Mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn trên mô hình SVM*

2.4.2. Đánh giá thuật toán đề xuất

Phần này tiến hành thực nghiệm VPOFRS_ARD và đánh giá mối quan hệ giữa kích thước tập rút gọn thu được tại mỗi giá trị β được đề xuất, trình bày mối quan



Hình 2.4: *Mối quan hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn trên mô hình k-NN*

hệ giữa kích thước và độ chính xác phân lớp của tập rút gọn. Trên có sở đó, xác đinh khoảng giá trị của β cho tập rút gọn có kích thước và độ chính xác phân lớp tốt nhất.

Hình 2.2 trình bày mối quan hệ giữa kích thước tập rút gọn và giá trị β (ngưỡng điều chỉnh) trong thuật toán đề xuất. Phân tích cho thấy, khoảng giá trị β tối ưu có sự khác biệt đáng kể giữa các bộ dữ liệu, tạo ra thách thức trong việc lựa chọn một giá trị phù hợp cho từng trường hợp.

Nhìn chung, khi giá trị β tăng lên, kích thước của tập rút gọn có xu hướng giảm dần, cho thấy mối tương quan nghịch giữa hai yếu tố này. Tuy nhiên, bộ dữ liệu UFDC lại thể hiện một xu hướng trái ngược, với kích thước tập rút gọn tăng lên khi β tăng. Sự khác biệt này nhấn mạnh sự phụ thuộc của thuật toán vào đặc tính riêng của từng bộ dữ liệu và tầm quan trọng của việc điều chỉnh tham số β một cách cẩn thận để đạt được kết quả tốt nhất. Việc lựa chọn β thích hợp đòi hỏi sự cân nhắc kỹ lưỡng để đảm bảo tập rút gọn thu được vừa nhỏ gọn vừa giữ được thông tin quan trọng.

Hình 2.3 thể hiện sự ảnh hưởng của tham số β đến độ chính xác phân lớp của mô hình SVM sau khi áp dụng thuật toán rút gọn thuộc tính. Khi so sánh với Hình 2.2, có thể thấy rõ mối tương quan giữa kích thước tập thuộc tính và khả năng dự đoán của mô hình. Thông thường, việc loại bỏ thuộc tính (giảm kích thước tập rút gọn) sẽ làm giảm độ chính xác, do mất mát thông tin.

Tuy nhiên, đáng chú ý là trên hai bộ dữ liệu UFDC và SHDC, độ chính xác phân lớp lại không hề suy giảm khi β thay đổi, cho thấy thuật toán đã loại bỏ thành công những thuộc tính ít hoặc không mang lại giá trị thông tin. Thậm chí, với bộ UFDC, độ chính xác còn tăng vọt từ 44% lên 57%. Điều này chứng tỏ, thuật toán không chỉ loại bỏ thuộc tính dư thừa mà còn có khả năng loại bỏ hiệu quả các thuộc tính gây nhiễu (noisy features), từ đó giúp mô hình SVM khái quát hóa tốt hơn và đưa ra dự đoán chính xác hơn. Đây là một kết quả quan trọng, cho thấy tiềm năng của thuật toán trong việc cải thiện hiệu suất phân loại trên các bộ dữ liệu phức tạp, nhiều nhiễu.

Tương tự như phân tích trên mô hình SVM, Hình 2.4 cho thấy mối quan hệ giữa tham số β, kích thước tập rút gọn và độ chính xác phân lớp khi sử dụng mô hình

k-NN. Xu hướng chung vẫn là độ chính xác giảm khi kích thước tập thuộc tính giảm đi, phản ánh sự mất mát thông tin tiềm năng. Tuy nhiên, bộ dữ liệu **UFDC** tiếp tục là một ngoại lệ đáng chú ý. Sau khi áp dụng thuật toán rút gọn thuộc tính và điều chỉnh tham số β, độ chính xác phân lớp trên bộ UFDC đã tăng vọt từ 70% lên đến 86%. Kết quả này khẳng định lại khả năng của thuật toán trong việc xác định và loại bỏ các thuộc tính gây nhiễu, từ đó giúp mô hình k-NN hoạt động hiệu quả hơn, đặc biệt là trên những bộ dữ liệu vốn có độ chính xác ban đầu thấp.

2.4.3. So sánh các thuật toán rút gọn thuộc tính cải thiện độ chính xác phân lớp

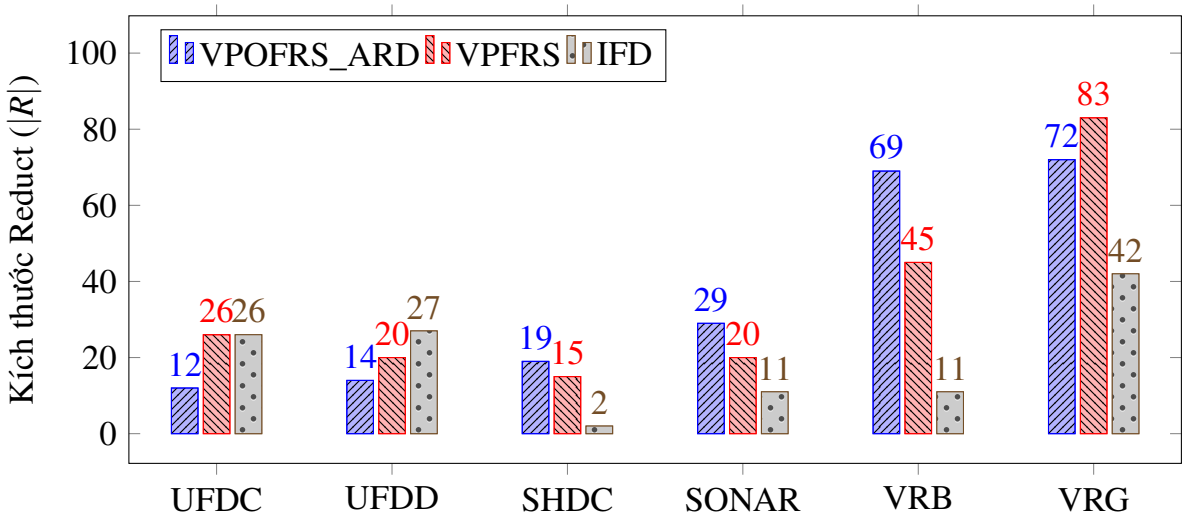
Sau khi xác định khoảng giá trị β tối ưu cho từng bộ dữ liệu, cho phép lựa chọn ra tập rút gọn có sự cân bằng tốt nhất giữa kích thước và độ chính xác phân lớp, bước tiếp theo là so sánh hiệu năng của thuật toán đề xuất với các thuật toán VPFRS và IFD hiện có. Ba tiêu chí chính được sử dụng để đánh giá và so sánh các thuật toán bao gồm: thời gian thực hiện, kích thước tập rút gọn và độ chính xác phân lớp.

2.4.3.1. So sánh kích thước của tập rút gọn thu được từ các thuật toán

Hình 2.5 so sánh kích thước tập rút gọn giữa các thuật toán khác nhau. Kết quả cho thấy IFD có xu hướng tạo ra tập rút gọn với kích thước trung bình nhỏ nhất trên các bộ dữ liệu. Tuy nhiên, điều thú vị là hai bộ dữ liệu UFDC và UFDD lại cho thấy sự vượt trội của thuật toán VPOFRS_ARD.

Điều này đáng chú ý bởi vì UFDC và UFDD là hai bộ dữ liệu có độ chính xác phân lớp ban đầu thấp nhất, cho thấy sự hiện diện của nhiễu trong dữ liệu. Tuy nhiên, thuật toán VPOFRS_ARD đã tạo ra các tập rút gọn không chỉ nhỏ hơn mà còn đạt độ chính xác phân lớp cao hơn so với các thuật toán khác. Điều này ngụ ý rằng VPOFRS_ARD có khả năng xác định và loại bỏ các thuộc tính nhiễu một cách hiệu quả hơn, cho phép mô hình phân loại tập trung vào các thông tin quan trọng và cải thiện hiệu suất dự đoán. Như vậy, mặc dù IFD có xu hướng tạo ra tập rút gọn nhỏ nhất, nhưng VPOFRS_ARD cho thấy tiềm năng lớn hơn trong việc xử lý dữ liệu nhiễu và cải thiện

độ chính xác phân lớp.

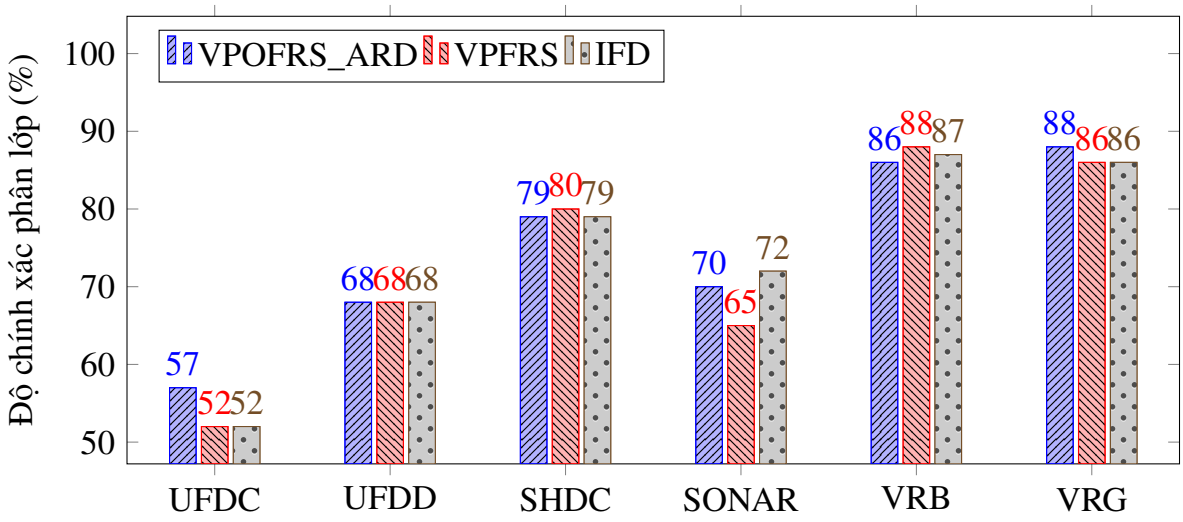


Hình 2.5: *So sánh kích thước tập rút gọn thu được từ các thuật toán*

2.4.3.2. So sánh độ chính xác phân lớp của tập rút gọn thu được từ các thuật toán

Hình 2.6 so sánh độ chính xác phân lớp (trên mô hình SVM) của các tập rút gọn do các thuật toán tạo ra. Kết quả cho thấy độ chính xác giữa các thuật toán phần lớn không khác biệt nhiều, gợi ý rằng tất cả đều bảo toàn tốt khả năng phân loại tổng thể. Dù vậy, bộ dữ liệu **UFDC** lại nổi lên như một trường hợp đặc biệt, khi tập rút gọn từ VPOFRS_ARD đạt được độ chính xác cao hơn đáng kể so với IFD và VPFRS.

Khi xem xét đồng thời với Hình 2.5 (so sánh kích thước tập rút gọn), nhận thấy rằng VPOFRS_ARD không chỉ nâng cao độ chính xác trên UFDC (tăng 5% so với VPFRS) mà còn giảm kích thước tập rút gọn xuống chỉ còn một nửa. Điều này cho thấy VPOFRS_ARD đã thành công trong việc loại bỏ các thuộc tính không liên quan hoặc gây nhiễu, giúp mô hình SVM tập trung vào những yếu tố thực sự quan trọng, từ đó cải thiện hiệu suất. Mặc dù IFD tiếp tục thể hiện khả năng tạo ra các tập rút gọn nhỏ hơn (với độ chính xác tương đương), kết quả của VPOFRS_ARD trên UFDC nhấn mạnh tiềm năng của thuật toán trong việc vừa giảm chiều dữ liệu, vừa tăng cường khả năng dự đoán của mô hình, một yếu tố then chốt trong các ứng dụng thực tế.



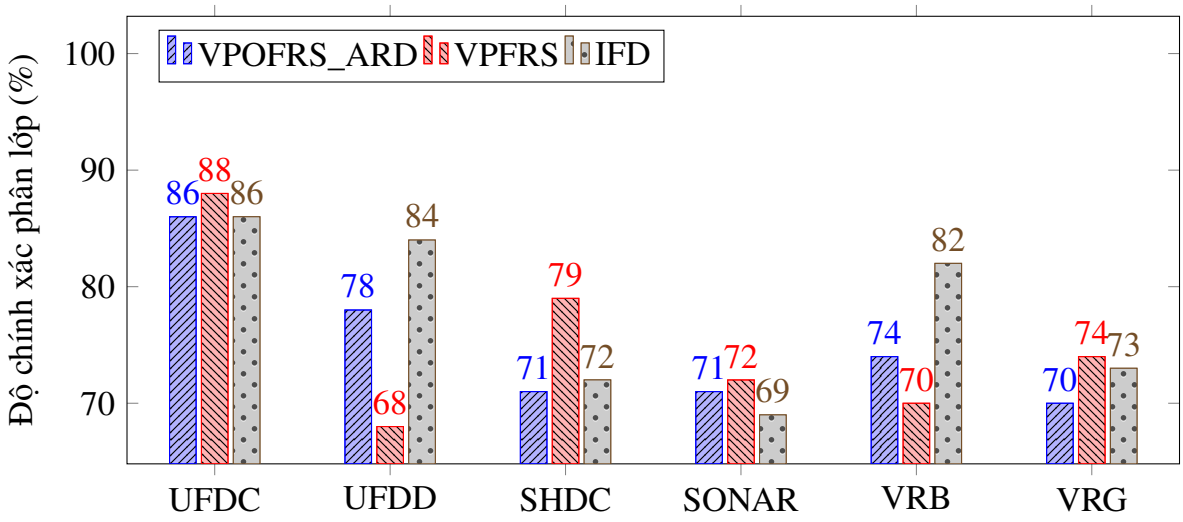
Hình 2.6: *So sánh độ chính xác phân lớp của các tập rút gọn xây dựng trên mô hình phân lớp SVM*

Hình 2.7 so sánh độ chính xác phân lớp (mô hình k-NN) của các tập rút gọn từ các thuật toán. Kết quả cho thấy, tương tự như với mô hình SVM, sự khác biệt về độ chính xác giữa các thuật toán thường không đáng kể. Tuy nhiên, bộ dữ liệu **UFDC** tiếp tục chứng tỏ là một trường hợp đặc biệt, khi VPOFRS_ARD mang lại độ chính xác phân lớp vượt trội so với IFD và VPFRS.

Khi kết hợp với thông tin về kích thước tập rút gọn (Hình 2.5), thấy rằng VPOFRS_ARD, trên bộ **UFDC**, không chỉ giảm số lượng thuộc tính xuống một nửa so với VPFRS mà còn tăng độ chính xác thêm 5%. Điều này càng củng cố nhận định rằng VPOFRS_ARD có khả năng loại bỏ hiệu quả nhiễu trong dữ liệu, giúp mô hình k-NN khái quát hóa tốt hơn.

2.4.3.3. So sánh thời gian thực hiện của các thuật toán

Như đã đề cập trong phần khảo sát các phương pháp rút gọn thuộc tính cải thiện độ chính xác phân lớp trên các bộ dữ liệu nhiễu. Phương pháp IFD là tiếp cận có thời gian tính toán gấp đôi so với các tiếp cận tính toán trên không gian xấp xỉ mờ. Do đó, phần này chỉ so sánh thời gian thực hiện của thuật toán đề xuất so với thuật toán



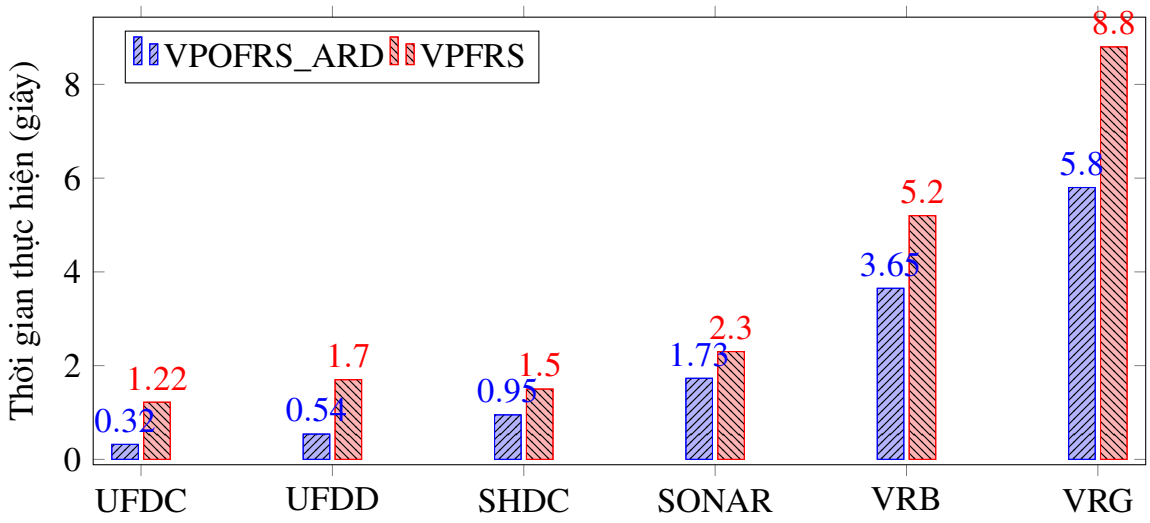
Hình 2.7: *So sánh độ chính xác phân lớp của các tập rút gọn xây dựng trên mô hình phân lớp k-NN*

VPFRS.

Hình 2.8 so sánh thời gian thực hiện của thuật toán đề xuất VPOFRS_ARD với thuật toán VPFRS gốc trên sáu bộ dữ liệu khác nhau. Nhìn chung, VPOFRS_ARD thể hiện sự vượt trội về hiệu quả thời gian, hoàn thành quá trình rút gọn thuộc tính nhanh hơn VPFRS trên mọi bộ dữ liệu thử nghiệm.

Mức độ cải thiện thời gian đáng chú ý nhất trên các bộ dữ liệu lớn hơn như VRB và VRG, cho thấy lợi ích của VPOFRS_ARD trong việc xử lý các bộ dữ liệu phức tạp hơn. Ví dụ, trên VRG, VPOFRS_ARD hoàn thành trong 5.8 giây, trong khi VPFRS mất đến 8.8 giây. Sự khác biệt này có thể là do các cải tiến của mô hình VPOFRS, giảm thiểu số lượng tính toán cần thiết để xác định reduct.

Tuy nhiên, trên các bộ dữ liệu nhỏ hơn như UFDC và UFDD, sự khác biệt về thời gian là ít rõ rệt hơn. Điều này có thể là do chi phí cố định liên quan đến việc khởi động thuật toán. Tóm lại, kết quả phân tích cho thấy VPOFRS_ARD là một thuật toán hứa hẹn để rút gọn thuộc tính hiệu quả trên các bộ dữ liệu có chứa nhiễu, đặc biệt là khi đối mặt với các bộ dữ liệu lớn và phức tạp.



Hình 2.8: *So sánh thời gian thực hiện giữa các thuật toán*

2.5. Kết luận Chương 2

Chương 2, luận án trình bày tiếp cận đặc biệt hữu ích trong việc giải quyết các vấn đề liên quan đến dữ liệu không chắc chắn và nhiễu, vốn là những thách thức phổ biến trong các bài toán phân tích dữ liệu thực tế.

Điểm cốt lõi của phương pháp nằm ở việc xây dựng một độ đo để đánh giá mức độ quan trọng của từng thuộc tính. Độ đo này dựa trên độ phụ thuộc, được định nghĩa theo cách tiếp cận VPOFRS, cho phép xác định chính xác mức độ ảnh hưởng của mỗi thuộc tính đến quá trình phân loại dữ liệu. Điều này giúp chọn ra những thuộc tính quan trọng nhất, đồng thời loại bỏ những thuộc tính ít liên quan hoặc gây nhiễu.

Ngoài việc áp dụng cho bài toán rút gọn thuộc tính, luận án cũng chỉ ra rằng mô hình tập thô VPOFRS có tiềm năng ứng dụng rộng rãi trong các bài toán phân tích dữ liệu khác. Với khả năng xử lý dữ liệu mờ và không chắc chắn, VPOFRS có thể trở thành một công cụ mạnh mẽ trong nhiều lĩnh vực khác nhau, từ nhận dạng mẫu đến dự báo và ra quyết định, mở ra nhiều hướng nghiên cứu và ứng dụng trong tương lai.

CHƯƠNG 3. PHƯƠNG PHÁP TÍNH TOÁN GIA TĂNG TẬP RÚT GỌN THEO TIẾP CẬN TÍNH TOÁN HẠT THÔNG TIN TRÊN KHÔNG GIAN XẤP XỈ MỜ

Chương 3 của luận án tập trung vào việc giải quyết bài toán chọn lọc thuộc tính gia tăng trong môi trường dữ liệu động, cụ thể là khi bảng quyết định có sự thay đổi về tập đối tượng. Xuất phát từ những hạn chế của các phương pháp truyền thống, chủ yếu dựa trên độ đo khoảng cách giữa các phân hoạch, trong việc xử lý hiệu quả dữ liệu biến động. Luận án đã đề xuất một hướng tiếp cận mới dựa trên việc mở rộng khái niệm hạt thông tin tri thức để xây dựng độ đo trên không gian xấp xỉ mờ.

Điểm nhấn chính của chương là việc xây dựng các công thức tính toán gia tăng cho phép cập nhật tập rút gọn một cách nhanh chóng và hiệu quả khi có sự bổ sung hoặc loại bỏ đối tượng trong bảng quyết định. Cụ thể, hai thuật toán FIGAMO (Fuzzy Information Granularity-based Algorithm for Adding Objects) và FIGDMO (Fuzzy Information Granularity-based Algorithm for Deleting Objects) đã được phát triển, tương ứng với hai kịch bản thay đổi dữ liệu.

Các kết quả thực nghiệm cho thấy, thuật toán FIGAMO mang lại sự cải thiện đáng kể về thời gian tính toán so với các phương pháp trước đây. Đặc biệt, trên một số bộ dữ liệu có số mẫu lớn, thuật toán FIGAMO không chỉ giúp giảm thiểu số lượng thuộc tính mà còn đồng thời nâng cao độ chính xác phân lớp. Điều này chứng minh rằng, việc kết hợp khái niệm hạt thông tin tri thức với không gian xấp xỉ mờ có thể tạo ra các phương pháp chọn lọc thuộc tính gia tăng không chỉ hiệu quả về mặt tính toán mà còn có khả năng cải thiện chất lượng mô hình.

Tuy nhiên, chương 3 cũng chỉ ra rằng việc lựa chọn điều kiện dừng phù hợp đóng vai trò quan trọng trong việc cân bằng giữa thời gian tính toán và kích thước của tập rút gọn. Nhìn chung, chương 3 đã đóng góp một phương pháp tiếp cận đầy hứa hẹn

cho việc giải quyết bài toán chọn lọc thuộc tính gia tăng trong môi trường dữ liệu động, đặc biệt là với những bộ dữ liệu có số mẫu lớn.

Các kết quả nghiên cứu được công bố trong các công trình [CT1, CT4]. Trước khi đi vào nội dung chính, phần kiến thức cơ sở về không gian xấp xỉ mờ, ma trận quan hệ mờ và một số tính chất được nhắc lại.

3.1. Các kiến thức cơ sở

Phần này sẽ trình bày một số các kiến thức cơ sở về ma trận quan hệ tương đương mờ của một thuộc tính và ma trận quan hệ tương đương mờ của tập thuộc tính. Các ma trận quan hệ này được sử dụng để xây dựng các công cụ tính toán trong phần tiếp theo của chương nghiên cứu. Chi tiết các kiến thức cơ sở này có thể tìm hiểu tại các công trình [74], [76].

Định nghĩa 3.1 (Quan hệ tương tự của hai đối tượng trên một thuộc tính)**.** Cho bảng quyết định $NDT =< U, C, D >$, quan hệ tương tự của hai đối tượng $i, j \in U$ theo thuộc tính $a \in C$ được xác định bởi công thức sau:

$$s_{ij}^{a} = 1 - |a(i) - a(j)| \tag{3.1}$$

Rõ ràng công thức (3.1) là một quan hệ tương đương mờ do:

(1) Tính phản xạ: $s_{ij}^{a} = 1$ nếu $i = j$;

(2) Tính đối tướng: $s_{ij}^{a} = s_{ji}^{a}$;

(3) Tính bắc cầu: $s_{ik}^{a} \geq min(s_{ij}^{a}, s_{jk}^{a})$.

Định nghĩa 3.2 (Ma trận quan hệ tương đương mờ của một thuộc tính)**.** Cho bảng quyết định $NDT =< U, C, D >$, quan hệ tương tự giữa các đối tượng của U theo thuộc

tính $a \in C \cup D$ có thể được biểu diễn bởi ma trận quan hệ mờ như sau:

$$S(a,U) = \begin{bmatrix} s_{11}^a & s_{12}^a & \dots & s_{1n}^a \\ s_{21}^a & s_{22}^a & \dots & s_{2n}^a \\ \dots & \dots & \dots & \dots \\ s_{n1}^a & s_{n2}^a & \dots & s_{nn}^a \end{bmatrix} \tag{3.2}$$

Trong đó, $n = |U|$, $s_{ij}^a \in [0,1] \forall i, j \in [1,n]$ là quan hệ tương tự giữa đối tượng i và đối tượng j trong U trên thuộc tính a.

Định nghĩa 3.3 (Ma trận quan hệ tương đương mờ của một tập thuộc tính)**.** Cho bảng quyết định $NDT =< U, C, D >$, ma trận quan hệ của các đối tượng trong U trên tập thuộc tính $P \subseteq C$ được xác định như sau:

$$S(P,U) = \bigcap_{a \in P} S(a,U) \tag{3.3}$$

Trong đó, $s_{ij}^P = min\{s_{ij}^a : a \in P\}$.

3.2. Độ đo hạt thông tin mờ

Phần này giới thiệu phương pháp mở rộng độ đo hạt thông tin trên không gian xấp xỉ rõ sang độ đo hạt thông tin trên không gian xấp xỉ mờ, gọi tắt là độ đo hạt thông tin mờ. Trước tiên, khái niệm hạt thông tin mờ của thuộc tính được định nghĩa trước, sau đó các độ đo trên hạt thông tin mờ được xây dựng. Cuối cùng là thuật toán rút gọn thuộc tính theo tiếp cận độ đo hạt thông tin mờ cho bảng quyết định NDT được đề xuất.

3.2.1. Hạt thông tin mờ

Định nghĩa 3.4 (Hạt thông tin mờ)**.** Cho bảng quyết định $NDT =< U, C, D >$ và tập thuộc tính $P \subseteq C$, khi đó hạt thông tin mờ của P trên U được xác định bởi:

$$FIG(P,U) = 1 - \frac{\sum_{i=1}^{|U|} \left|[i]_P^U\right|}{|U|^2} \tag{3.4}$$

Trong đó $\left|[i]_P^U\right| = \sum_{j=1}^{|U|} s_{ij}^P$

Ví dụ: Xét ma trận quan hệ của thuộc tính $a \in C$ trên tập đối tượng U:

$$R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}.$$

Khi đó, hạt thông tin mờ của thuộc tính a theo công thức (3.4) được tính như sau:

$FIG(a,U) = 1 - \left(\frac{|[1 \quad 1 \quad 0.8 \quad 0.2 \quad 0.2 \quad 0.2]| + ... + |[0.2 \quad 0.2 \quad 0.4 \quad 1 \quad 1 \quad 1]|}{|6|^2}\right)$

$= 1 - \frac{22}{36} = \frac{14}{36}$

Mệnh đề 3.1 (Tính đơn điệu của hạt thông tin mờ)**.** *Cho bảng quyết định* $NDT =< U, C, D >$*, nếu* $P \subseteq Q \subseteq C$ *thì* $FIG(P,U) \leq FIG(Q,U)$.

Chứng minh. Dựa theo công thức (3.3), ta có, với mọi $i, j \in [1, |U|]$, nếu $P \subseteq Q$ thì $s_{ij}^Q \leq s_{ij}^P$. Khi đó, ta có: $\sum_{i=1}^{|U|} \left|[i]_Q^U\right| \leq \sum_{i=1}^{|U|} \left|[i]_P^U\right|$, do đó $\frac{\sum_{i=1}^{|U|} \left|[i]_Q^U\right|}{|U|^2} \leq \frac{\sum_{i=1}^{|U|} \left|[i]_P^U\right|}{|U|^2}$. Dựa trên định nghĩa (3.1), ta có: $1 - \frac{\sum_{i=1}^{|U|} \left|[i]_P^U\right|}{|U|^2} \leq 1 - \frac{\sum_{i=1}^{|U|} \left|[i]_Q^U\right|}{|U|^2}$ hay $FIG(P,U) \leq FIG(Q,U)$. Ta có điều phải chứng minh. □

3.2.2. Độ đo hạt thông tin mờ

Định nghĩa 3.5 (Độ đo độ phụ thuộc của hạt thông tin mờ)**.** Cho bảng quyết định $NDT =< U,C,D >$ và $P \subseteq C$, khi đó hạt thông tin mờ của D phụ thuộc vào P trên U được xác định bởi:

$$FIG(D|P,U) = FIG(D \cup P,U) - FIG(P,U) \tag{3.5}$$

Mệnh đề 3.2 (Tính đơn điệu của độ đo độ phụ thuộc của hạt thông tin mờ)**.** *Cho bảng quyết định $NDT =< U,C,D >$, nếu $P \subseteq Q \subseteq C$ thì $FIG(D|P,U) \leq FIG(D|Q,U)$.*

Chứng minh. Dựa theo công thức (3.3), ta có, với mọi $i, j \in [1,|U|]$, nếu $P \subseteq Q$ thì $s_{ij}^{Q \cup D} \leq s_{ij}^{P \cup D}$.

Khi đó, ta có: $\sum_{i=1}^{|U|} \left| [i]_{Q \cup D}^{U} \right| \leq \sum_{i=1}^{|U|} \left| [i]_{P \cup D}^{U} \right|$, do đó $\frac{\sum_{i=1}^{|U|} \left| [i]_{Q \cup D}^{U} \right|}{|U|^2} \leq \frac{\sum_{i=1}^{|U|} \left| [i]_{P \cup D}^{U} \right|}{|U|^2}$.

Dựa trên định nghĩa (3.4), ta có: $1 - \frac{\sum_{i=1}^{|U|} \left| [i]_{P \cup D}^{U} \right|}{|U|^2} \leq 1 - \frac{\sum_{i=1}^{|U|} \left| [i]_{Q \cup D}^{U} \right|}{|U|^2}$ hay $FIG(P \cup D,U) \leq FIG(Q \cup D,U)$. Ta có điều phải chứng minh. □

3.2.3. Xây dựng thuật toán rút gọn thuộc tính

Mệnh đề 3.2 cho thấy sự đơn điệu của hạt thông tin mờ. Nếu tập thuộc tính P tăng về số lượng thì độ phụ thuộc của D vào P cũng tăng và ngược lại. Dựa trên tính chất đơn điệu của hạt thông tin mờ đề xuất, độ đo độ quan trọng của thuộc tính và cấu trúc tập rút gọn lần lượt được định nghĩa như sau:

Định nghĩa 3.6 (Độ quan trọng của thuộc tính)**.** Cho bảng quyết định $NDT =< U,C,D >$ và $B \subseteq C$, độ quan trọng của thuộc tính $a \in C - B$ được xác định như sau:

$$Sig_{D|B}^{U}(a) = FIG(D|B \cup \{a\},U) - FIG(D|B,U) \tag{3.6}$$

Định nghĩa 3.7 (Cấu trúc tập rút gọn)**.** Cho bảng quyết định $NDT =< U,C,D >$ và $B \subseteq C$, khi đó B được gọi là tạp rút gọn của C nếu:

1)$FIG(D|B,U) = FIG(D|C,U)$

2)$\forall b \in B : FIG(D|B-\{b\},U) \neq FIG(D|C,U)$

Dựa trên các định nghĩa về độ quan trọng của thuộc tính và tập rút gọn theo hạt thông tin mờ đã đề xuất, thuật toán rút gọn thuộc tính theo tiếp cận tính toán hạt thông tin mờ (Fuzzy Information Granularity - FIG) được đề xuất. Thuật toán FIG được sử dụng để tìm tập rút gọn khi bảng quyết định NDT không thay đổi. Các bước chính của thuật toán FIG gồm có: bước 1, khởi tạo tập rút gọn $S = \emptyset$, bước 3 đến bước 5 thực hiện các bước lặp tính toán $Sig^U_{D|S}(a)$ dựa trên định nghĩa 3.6, bước 6 cho biết thuộc tính s là thuộc tính có độ quan trọng lớn nhất, bước 7 bổ sung thuộc tính s vào tập rút gọn S. Quá trình này được lặp lại cho đến khi điều kiện dừng thỏa mãn theo định nghĩa 3.7. Bước 9 ta thu được tập rút gọn trên U, kí hiệu là R_U. Sau đây là nội dung chi tiết của thuật toán FIG.

Thuật toán 3.1 Thuật toán rút gọn thuộc tính theo tiếp cận hạt thông tin mờ FIG

Input: $NDT =< U,C,D >$
Output: Tập rút gọn R_U

1: $S \leftarrow \emptyset$;
2: **while** $FIG(D|C,U) - FIG(D|S,U) > 0.5\%$ **do**
3: **for all** $a \in (C-S)$ **do**
4: **compute** $Sig^U_{D|S}(a)$
5: **end for**
6: $s = max\left\{Sig^U_{D|S}(a)\right\}$
7: $S \leftarrow S \cup \{s\}$
8: **end while**
9: $R_U \leftarrow S$
10: **return** R_U

Độ phức tạp của thuật toán FIG được phân tích dựa trên các bước chính từ 3-5, độ phức tạp của các bước này là: $O\left(|U|^2|C||S|\right)$, trong trường hợp tồi nhất, phải duyệt hết các thuộc tính, độ phức tạp là $O\left(|U|^2|C|^2\right)$, tương tự như thuật toán FD [76]. Do đó, độ phức tạp của thuật toán FIG là $O\left(|U|^2|C|^2\right)$.

Tuy nhiên, trong thực tế để tối ưu kích thước tập rút gọn thu được, ta chỉ lấy tập rút gọn xấp xỉ theo tập thuộc tính gốc. Với các bộ dữ liệu có kích thước vừa và nhỏ, thuật

toán sử dụng mức độ chênh lệch là 0.5 giữa hạt thông tin mờ của tập thuộc tính gốc với tập rút gọn thu được. Rõ ràng mức độ chênh lệch càng cao, kích thước tập rút gọn thu được càng nhỏ, thời gian thực hiện của thuật toán FIG càng nhanh và ngược lại.

3.2.4. Ví dụ số minh họa thuật toán

Để làm rõ tính đúng đắn của các công thức và thuật toán đề xuất. Phần này sẽ trình bày chi tiết các bước thực hiện của thuật toán qua từng công thức. Cụ thể như sau:

Đầu vào: Bảng quyết định $DT = (U, C, D)$, $\beta = 0.5$.

Đầu ra: Tập rút gọn B.

Bước 1: Khởi tạo $B = \emptyset$, $R_B = [1]_{|U|\times|U|}$. Trong đó, $[1]_{|U|\times|U|}$ là ma trận vuông kích thước $|U| \times |U|$ với mọi phần tử của ma trận có giá trị là 1.

Bước 2: Tính độ phụ thuộc của thuộc tính quyết định D vào tập thuộc tính B và tập thuộc tính C. Trước tiên ta áp dụng công thức (2.10) ta tính ma trận quan hệ của từng thuộc tính trong C như sau:

$$R_a = \begin{bmatrix} 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 1 & 1 & 0.8 & 0.2 & 0.2 & 0.2 \\ 0.8 & 0.8 & 1 & 0.4 & 0.4 & 0.4 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \\ 0.2 & 0.2 & 0.4 & 1 & 1 & 1 \end{bmatrix}; R_b = \begin{bmatrix} 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 1 & 1 & 0.8 & 0.8 & 0.6 & 0.6 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.8 & 0.8 & 1 & 1 & 0.8 & 0.8 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \\ 0.6 & 0.6 & 0.8 & 0.8 & 1 & 1 \end{bmatrix}$$

$$R_c = \begin{bmatrix} 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 1 & 0.2 & 0.4 & 0.4 & 1 \\ 0.8 & 0.2 & 1 & 0.8 & 0.8 & 0.2 \\ 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 1 & 0.4 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 1 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}; R_d = \begin{bmatrix} 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 0.8 & 1 & 0.6 & 0.8 & 0.8 & 0.6 \\ 0.8 & 0.6 & 1 & 0.8 & 0.8 & 0.2 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 1 & 0.8 & 0.8 & 1 & 1 & 0.4 \\ 0.4 & 0.6 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}$$

$$R_e = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.6 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 1 & 0.4 & 0.4 & 0.4 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \end{bmatrix}; R_f = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.6 & 0.8 & 0.8 & 0.8 \\ 0.6 & 0.6 & 1 & 0.4 & 0.4 & 0.4 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \\ 0 & 0.8 & 0.4 & 1 & 1 & 1 \end{bmatrix}$$

Áp dụng công thức (1.8) và phép toán T thứ nhất trong bảng 1.8 để xây dựng ma trận quan hệ của tập thuộc tính C như sau:

$$R_C = R_a \cap R_b \cap ... \cap R_f = \begin{bmatrix} 1 & 0.2 & 0.6 & 0 & 0 & 0 \\ 0.2 & 1 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.6 & 0.2 & 1 & 0.4 & 0.4 & 0.2 \\ 0 & 0.2 & 0.4 & 1 & 0.8 & 0.4 \\ 0 & 0.2 & 0.4 & 0.8 & 1 & 0.4 \\ 0 & 0.2 & 0.2 & 0.4 & 0.4 & 1 \end{bmatrix}$$

Ma trận quan hệ của thuộc tính quyết định D tương ứng là: $R_D = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$

Áp dụng công thức (3.5) ta tính hạt thông tin phụ thuộc của thuộc tính D tương ứng với tập thuộc tính B và C như sau:

$FIG(D|C,U) = 1 - \frac{10}{36} = \frac{26}{36}$;

$FIG(D|B,U) = 1 - \frac{36}{36} = 0$.

Bước 3: Vì $\left(FIG(D|C,U) = 1 - \frac{10}{36} = \frac{26}{36} - FIG(D|B,U) = 1 - \frac{36}{36} = 0\right) > 0.5\%$, thực hiện các bước trong vòng lặp ta có:

Áp dụng công thức (3.6):

- $SIG^U_{D|B}(a) = FIG(D|B \cup \{a\}, U) - FIG(D|B,U) = \frac{10.4}{36} - 0 = \frac{10.4}{36}$

- $SIG^{U}_{D|B}(b) = FIG(D|B \cup \{b\}, U) - FIG(D|B,U) = \frac{14.8}{36} - 0 = \frac{14.8}{36}$

- $SIG^{U}_{D|B}(c) = FIG(D|B \cup \{c\}, U) - FIG(D|B,U) = \frac{12}{36} - 0 = \frac{12}{36}$

- $SIG^{U}_{D|B}(d) = FIG(D|B \cup \{d\}, U) - FIG(D|B,U) = \frac{12.8}{36} - 0 = \frac{12.8}{36}$

- $SIG^{U}_{D|B}(e) = FIG(D|B \cup \{e\}, U) - FIG(D|B,U) = \frac{13.2}{36} - 0 = \frac{13.2}{36}$

- $SIG^{U}_{D|B}(f) = FIG(D|B \cup \{f\}, U) - FIG(D|B,U) = \frac{13.2}{36} - 0 = \frac{13.2}{36}$

Vì $SIG^{U}_{D|B}(b)$ là lớn nhất. Do đó, bổ sung thuộc tính b vào tập thuộc tính B ta có: $B = B \cup \{b\} = \{b\}$.

Vì $\left(FIG(D|C,U) = \frac{26}{36} - FIG(D|B,U) = \frac{14.8}{36}\right) > 0.5\%$. Tiếp tục thực hiện vòng lặp ta có:

- $SIG^{U}_{D|B}(a) = FIG(D|B \cup \{a\}, U) - FIG(D|B,U) = 1 - \frac{11.2}{36} = \frac{24.8}{36} - \frac{14.8}{36} = \frac{10}{36}$

- $SIG^{U}_{D|B}(c) = FIG(D|B \cup \{c\}, U) - FIG(D|B,U) = 1 - \frac{12}{36} - \frac{14.8}{36} = \frac{9.2}{36}$

- $SIG^{U}_{D|B}(d) = FIG(D|B \cup \{d\}, U) - FIG(D|B,U) = 1 - \frac{13.2}{36} - \frac{14.8}{36} = \frac{8}{36}$

- $SIG^{U}_{D|B}(e) = FIG(D|B \cup \{e\}, U) - FIG(D|B,U) = 1 - \frac{12.4}{36} - \frac{14.8}{36} = \frac{8.8}{36}$

- $SIG^{U}_{D|B}(f) = FIG(D|B \cup \{f\}, U) - FIG(D|B,U) = 1 - \frac{12.4}{36} - \frac{14.8}{36} = \frac{8.8}{36}$

Vì $SIG^{U}_{D|B}(a)$ là lớn nhất. Do đó, bổ sung thuộc tính a vào tập thuộc tính B ta có: $B = B \cup \{a\} = \{b,a\}$.

Vì $\left(FIG(D|C,U) = \frac{26}{36} - FIG(D|B,U) = \frac{24.8}{36}\right) > 0.5\%$. Tiếp tục thực hiện vòng lặp ta có:

- $SIG^{U}_{D|B}(c) = FIG(D|B \cup \{c\}, U) - FIG(D|B,U) = 1 - \frac{10.8}{36} - \frac{24.8}{36} = \frac{0.4}{36}$

- $SIG^{U}_{D|B}(d) = FIG(D|B \cup \{d\}, U) - FIG(D|B,U) = 1 - \frac{10.8}{36} - \frac{24.8}{36} = \frac{0.4}{36}$

- $SIG^{U}_{D|B}(e) = FIG(D|B \cup \{e\}, U) - FIG(D|B,U) = 1 - \frac{10.4}{36} - \frac{24.8}{36} = \frac{0.8}{36}$

- $SIG^{U}_{D|B}(f) = FIG(D|B \cup \{f\}, U) - FIG(D|B,U) = 1 - \frac{10.4}{36} - \frac{24.8}{36} = \frac{0.8}{36}$

Vì $SIG^{U}_{D|B}(e)$ là lớn nhất. Do đó, bổ sung thuộc tính e vào tập thuộc tính B ta có: $B = B \cup \{e\} = \{a,b,e\}$.

Vì $\left(FIG(D|C,U) = \frac{26}{36} - FIG(D|B,U) = \frac{25.6}{36}\right) > 0.5\%$. Tiếp tục thực hiện vòng lặp ta có:

- $SIG^U_{D|B}(c) = FIG(D|B \cup \{c\}, U) - FIG(D|B,U) = 1 - \frac{10}{36} - \frac{25.6}{36} = \frac{0.4}{36}$

- $SIG^U_{D|B}(d) = FIG(D|B \cup \{d\}, U) - FIG(D|B,U) = 1 - \frac{10}{36} - \frac{25.6}{36} = \frac{0.4}{36}$

- $SIG^U_{D|B}(f) = FIG(D|B \cup \{f\}, U) - FIG(D|B,U) = 1 - \frac{10.4}{36} - \frac{25.6}{36} = 0$

Vì $SIG^U_{D|B}(c)$ là lớn nhất. Do đó, bổ sung thuộc tính c vào tập thuộc tính B ta có: $B = B \cup \{c\} = \{b,a,e,c\}$.

Vì $\left(FIG(D|C,U) = \frac{26}{36} - FIG(D|B,U) = \frac{26}{36}\right) = 0 < 0.5\%$, kết thúc vòng lặp While ta có tập rút gọn $B = \{b,a,e,c\}$.

3.3. Xây dựng thuật toán cập nhật tập rút gọn khi bổ sung tập đối tượng

Phần này trình bày về phương pháp xây dựng thuật toán cập nhật tập rút gọn khi bảng quyết định NDT bổ sung tập đối tượng. Các bước chính gồm có: 1) xây dựng công thức tính toán gia tăng khi bổ sung một đối tượng, 2) xây dựng công thức tính toán gia tăng khi bổ sung tập đối tượng, 3) xây dựng thuật toán cập nhật tập rút gọn khi bổ sung tập đối tượng FIGAMO. Trước tiên là phần xây dựng công thức tính toán gia tăng khi bổ sung một đối tượng cho bảng quyết định NDT.

3.3.1. Công thức gia tăng khi bổ sung thêm một đối tượng

Phần này, luận án mở rộng độ đo hạt thông tin, ứng dụng tính toán gia tăng khi bổ sung thêm một đối tượng

Mệnh đề 3.3 (Hạt thông tin mờ gia tăng khi bổ sung một đối tượng)**.** *Cho bảng quyết định NDT $=< U,C,D >$ và tập thuộc tính $P \subseteq C$, u là đối tượng cần được bổ sung vào NDT. Khi đó hạt thông tin mờ của P trên $U \cup \{u\}$ được xác định như sau:*

$$FIG(P,U \cup \{u\}) = 1 - \frac{1}{|U \cup \{u\}|^2}\left(|U|^2 - |U|^2 FIG(P,U) + 2\sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| + 1\right) \tag{3.7}$$

Trong đó: $[i]_P^{\{u\}}$ là lớp tương đương của $\{u\}$ trên tập thuộc tính P.

Chứng minh. Dựa theo công thức (3.4), ta có:

$$FIG(P,U\cup\{u\}) = 1 - \frac{\sum_{i=1}^{|U\cup\{u\}|}\left|[i]_P^{U\cup\{u\}}\right|}{|U\cup\{u\}|^2} = 1 - \frac{\sum_{i=1}^{|U|}\left|[i]_P^{U}\right| + \sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| + \left|[u]_P^{U}\right| + \left|[u]_P^{\{u\}}\right|}{|U\cup\{u\}|^2}$$

Công thức (3.1) có tính chất đối xứng, do đó: $\sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| = \left|[u]_P^{U}\right|$ và $\left|[u]_P^{\{u\}}\right| = 1$.

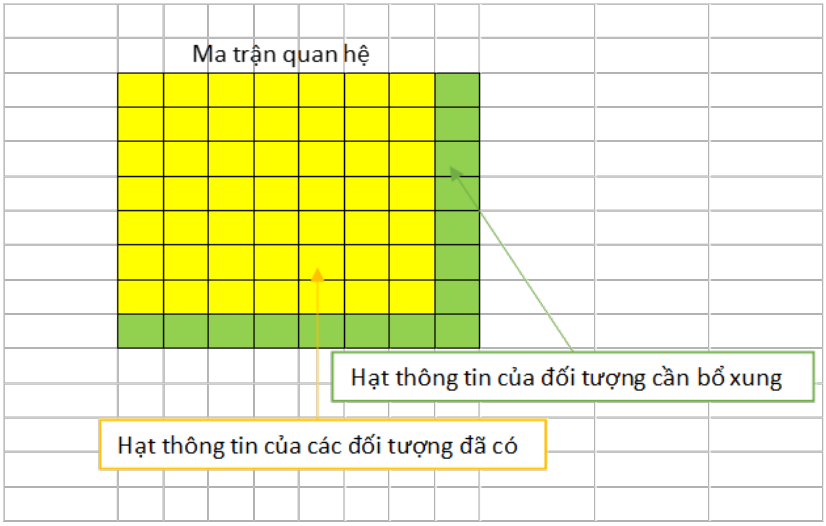
Khi đó:

$$FIG(P,U\cup\{u\}) = 1 - \frac{\sum_{i=1}^{|U|}\left|[i]_P^{U}\right| + 2\sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| + 1}{|U\cup\{u\}|^2} = 1 - \frac{|U|^2 - |U|^2 FIG(P,U) + 2\sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| + 1}{|U\cup\{u\}|^2}$$

$$= 1 - \frac{1}{|U\cup\{u\}|^2}\left(|U|^2 - |U|^2 FIG(P,U) + 2\sum_{i=1}^{|U|}\left|[i]_P^{\{u\}}\right| + 1\right).$$

Ta có điều phải chứng minh. □

Một cách khái quát, phương pháp tính toán gia tăng khi bổ sung một đối tượng có thể được minh họa như trong Hình 3.1



Hình 3.1: *Minh họa phương pháp gia tăng khi bổ sung một đối tượng*

3.3.2. Công thức gia tăng khi bổ sung thêm một tập đối tượng

Dựa trên công thức gia tăng khi bổ sung một đối tượng vào bảng quyết định NDT, công thức gia tăng được mở rộng cho trường hợp bổ sung nhiều đối tượng. Trên cơ sở đó, độ đo độ phụ thuộc của thuộc tính được đề xuất.

Mệnh đề 3.4 (Hạt thông tin mờ gia tăng khi bổ sung tập đối tượng)**.** *Cho bảng quyết định NDT* $=<U,C,D>$ *và tập thuộc tính* $P \subseteq Q \subseteq C$, $\Delta_u = \{u_1, u_2, ...u_k\}$ *là tập đối tượng cần được bổ sung vào NDT. Khi đó hạt thông tin mờ của P trên* $U \cup \Delta_u$ *được xác định như sau:*

$$FIG(P, U \cup \Delta_u) = 1 - \frac{1}{|U \cup \Delta_u|^2} \left(|U|^2 - |U|^2 FIG(P,U) + 2 \sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{\Delta_u} \right| \right) \tag{3.8}$$

Chứng minh. Dựa theo công thức (3.4), ta có:

$FIG(P, U \cup \Delta_u) = 1 - \frac{\sum_{i=1}^{|U \cup \Delta_u|} \left| [i]_P^{U \cup \Delta_u} \right|}{|U \cup \Delta_u|^2} = 1 - \frac{\sum_{i=1}^{|U|} \left| [i]_P^{U} \right| + \sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{U} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{\Delta_u} \right|}{|U \cup \Delta_u|^2}.$

Công thức (3.1) có tính chất đối xứng, do đó: $\sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| = \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{U} \right|$.

Khi đó: $FIG(P, U \cup \{u\}) = 1 - \frac{\sum_{i=1}^{|U|} \left| [i]_P^{U} \right| + 2 \sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{\Delta_u} \right|}{|U \cup \Delta_u|^2}$

$= 1 - \frac{|U|^2 - |U|^2 FIG(P,U) + 2 \sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{\Delta_u} \right|}{|U \cup \Delta_u|^2}$

$= 1 - \frac{1}{|U \cup \Delta_u|^2} \left(|U|^2 - |U|^2 FIG(P,U) + 2 \sum_{i=1}^{|U|} \left| [i]_P^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_P^{\Delta_u} \right| \right).$

Ta có điều phải chứng minh. □

Mệnh đề 3.5 (Hạt thông tin mờ phụ thuộc gia tăng khi bổ sung tập đối tượng)**.** *Cho bảng quyết định NDT* $=<U,C,D>$ *và tập thuộc tính* $P \subseteq Q \subseteq C$, $\Delta_u = \{u_1, u_2, ...u_k\}$ *là tập đối tượng cần được bổ sung vào NDT. Khi đó hạt thông tin mờ của D phụ thuộc vào P trên* $U \cup \Delta_u$ *được xác định như sau:*

$$FIG(D|P, U \cup \Delta_u) = \frac{1}{|U \cup \Delta_u|^2} \left(\begin{array}{l} |U|^2 FIG(D|P,U) - 2 \sum_{i=1}^{|U|} \left(\left| [i]_{D \cup P}^{\Delta_u} \right| - \left| [i]_P^{\Delta_u} \right| \right) \\ + \sum_{i=1}^{|\Delta_u|} \left(\left| [i]_{D \cup P}^{\Delta_u} \right| - \left| [i]_P^{\Delta_u} \right| \right) \end{array} \right) \tag{3.9}$$

Chứng minh. Dựa theo định nghĩa 3.5 và mệnh đề 3.4, ta có

$$FIG(D|P, U \cup \Delta_u) = FIG(D|P, U \cup \Delta_u) - FIG(P, U \cup \Delta_u)$$

$$= 1 - \frac{1}{|U \cup \Delta_u|^2} \left(|U|^2 - |U|^2 FIG(D \cup P, U) + 2 \sum_{i=1}^{|U|} \left| [i]_{D \cup P}^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_{D \cup P}^{\Delta_u} \right| \right)$$
$$- \left(1 - \frac{1}{|U \cup \Delta_u|^2} \left(|U|^2 - |U|^2 FIG(P, U) + 2 \sum_{i=1}^{|U|} \left| [i]_{P}^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_{P}^{\Delta_u} \right| \right) \right)$$
$$= \frac{1}{|U \cup \Delta_u|^2} \left(\begin{array}{l} |U|^2 FIG(D \cup P, U) - |U|^2 FIG(P, U) \\ - \left(2 \sum_{i=1}^{|U|} \left| [i]_{D \cup P}^{\Delta_u} \right| + \sum_{i=1}^{|\Delta_u|} \left| [i]_{D \cup P}^{\Delta_u} \right| - 2 \sum_{i=1}^{|U|} \left| [i]_{P}^{\Delta_u} \right| - \sum_{i=1}^{|\Delta_u|} \left| [i]_{P}^{\Delta_u} \right| \right) \end{array} \right)$$
$$= \frac{1}{|U \cup \Delta_u|^2} \left(|U|^2 FIG(D|P, U) - 2 \sum_{i=1}^{|U|} \left(\left| [i]_{D \cup P}^{\Delta_u} \right| - \left| [i]_{P}^{\Delta_u} \right| \right) + \sum_{i=1}^{|\Delta_u|} \left(\left| [i]_{D \cup P}^{\Delta_u} \right| - \left| [i]_{P}^{\Delta_u} \right| \right) \right)$$

Ta có điều phải chứng minh. □

3.3.3. Đề xuất thuật toán gia tăng khi bổ sung tập đối tượng

Trước khi đề xuất thuật toán, độ đo độ quan trọng của thuộc tính được mở rộng thông qua mệnh đề sau đây:

Mệnh đề 3.6 (Độ quan trọng của thuộc tính gia tăng bổ sung tập đối tượng)**.** *Cho bảng quyết định $NDT =< U, C, D >$ và $B \subseteq C$, độ quan trọng của thuộc tính $a \in C - B$ với B trên $U \cup \Delta_u$ được xác định như sau:*

$$Sig_{D|B}^{U \cup \Delta_u}(a) = \frac{1}{|U \cup \Delta_u|^2} \left(\begin{array}{l} |U|^2 Sig_{D|B}^{U}(a) + 2 \sum_{i=1}^{|U|} \left(\begin{array}{l} \left(\left| [i]_{D \cup B}^{\Delta_u} \right| - \left| [i]_{D \cup B \cup \{a\}}^{\Delta_u} \right| \right) \\ + \left(\left| [i]_{B}^{\Delta_u} \right| - \left| [i]_{B \cup \{a\}}^{\Delta_u} \right| \right) \end{array} \right) \\ + \sum_{i=1}^{|\Delta_u|} \left(\left(\left| [i]_{D \cup B}^{\Delta_u} \right| - \left| [i]_{D \cup B \cup \{a\}}^{\Delta_u} \right| \right) + \left(\left| [i]_{B}^{\Delta_u} \right| - \left| [i]_{B \cup \{a\}}^{\Delta_u} \right| \right) \right) \end{array} \right) \tag{3.10}$$

Chứng minh. Dựa theo định nghĩa 3.5 và mệnh đề 3.4, ta có

$$
\begin{aligned}
&FIG(D|B\cup\{a\},U\cup\Delta_u)-FIG(D|B,U\cup\Delta_u)\\
&=\frac{1}{|U\cup\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D|B\cup\{a\},U)-2\sum_{i=1}^{|U|}\left(\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|-\left|[u]^{\Delta_u}_{B\cup\{a\}}\right|\right)\end{array}\right)\\
&-\frac{1}{|U\cup\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D|B,U)-2\sum_{i=1}^{|U|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{B}\right|\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{B}\right|\right)\end{array}\right)\\
&=\frac{1}{|U\cup\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D|B\cup\{a\},U)-|U|^2FIG(D|B,U)\\+2\sum_{i=1}^{|U|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|+\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|+\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\end{array}\right)\\
&=\frac{1}{|U\cup\Delta_u|^2}\left(\begin{array}{l}|U|^2Sig^U_{D|B}(a)+2\sum_{i=1}^{|U|}\left(\begin{array}{l}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)\\+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\end{array}\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\right)\end{array}\right)
\end{aligned}
$$

Ta có điều phải chứng minh. □

Dựa trên mệnh đề 3.6. Thuật toán FIGAMO cập nhật tập rút gọn khi bổ sung tập đối tượng được đề xuất, sau đây là các bước chính của thuật toán FIGAMO. Bước 2, khởi tạo $S = R_U$, với R_U là tập rút gọn thu được trên bảng quyết định NDT với U đối tượng ban đầu. Bước 3, dựa theo mệnh đề 3.5, các biến F_s và F_c lưu các giá trị tương ứng của $FIG(D|S,U\cup\Delta_u)$ và $FIG(D|C,U\cup\Delta_u)$. Bước 4, nếu độ chênh lệch của F_s và F_c nhỏ hơn 0.5% thì trả lại S, nghĩa là tập rút gọn không thay đổi. Ngược lại, thực hiện các bước trong vòng lặp từ bước 8 đến 12 để cập nhật lại tập rút gọn S. Vòng lặp các bước từ 15 đến 17 nhằm loại bỏ các thuộc tính dư thừa nếu có. Bước 20 trả lại tập rút gọn được cập nhật của bảng quyết định NDT khi bổ sung Δ_u đối tượng.

Sau đây là phần đánh giá độ phức tạp của thuật toán FIFAMO. Bước 2 có độ phức tạp xấp xỉ là $O\left(|\Delta_u|^2|C|\right)$. Dựa trên mệnh đề 3.6, độ phức tạp các bước từ 6 đến 12 là $O\left(|\Delta_u|^2|C-S|^2\right)$. Các bước từ 13 đến 18 có độ phức tạp xấp xỉ là $O\left(|\Delta_u|^2|S|\right)$. Do

Thuật toán 3.2 Thuật toán rút gọn thuộc tính gia tăng bổ sung tập đối tượng theo tiếp cận hạt thông tin mờ FIGAMO

Input: $NDT =< U, C, D >$, R_U, Δ_u
Output: Tập rút gọn $R_{U\cup\Delta_u}$

$S \leftarrow R_U$;
$F_s = FIG(D|S, U\cup\Delta_u), F_c = FIG(D|C, U\cup\Delta_u)$;
if $|F_c - F_s| \leq 0.5\%$ **then**
  return S;
end if
while $|F_c - F_s| > 0.5\%$ **do**
  for all $a \in (C-S)$ **do**
    compute $Sig_{D|S}^{U\cup\Delta_u}(a)$
  end for
  $s = max\left\{Sig_{D|S}^{U\cup\Delta_u}(a)\right\}$
  $S \leftarrow S\cup\{s\}$
end while
for all $a \in S$ **do**
  $F_{s'} = FIG(D|S-\{a\}, U\cup\Delta_u)$;
  if $|F_c - F_{s'}| \leq 0.5\%$ **then**
    $S \leftarrow S-\{a\}$
  end if
end for
$R_{U\cup\Delta_u} \leftarrow S$
return $R_{U\cup\Delta_u}$

đó, độ phức tạp của thuật toán FIGAMO là $\max\left\{O\left(|\Delta_u|^2|C-S|^2\right), O\left(|\Delta_u|^2|C|\right)\right\}$, tương đương với thuật toán FDMAO [76].

Tuy nhiên, cũng giống như thuật toán FIG, thuật toán FIGAMO có thời gian thực hiện trong thực tế nhanh hơn thuật toán FDMAO do kích thước tập rút gọn được giới hạn theo độ lệch của hạt là 0.5% so với tập thuộc tính gốc.

Ví dụ 3.3. Theo ví dụ 3.2 ta thu được tập rút gọn của bảng quyết định 1.7 là $R_U = \{b, a, e, c\}$. Tiếp theo, chúng ta sẽ cập nhật lại tập rút gọn khi bổ sung đối tượng mới $u_7 = \{\frac{0.6,0.4,0.8,0.2,0.6,0.4}{C}, \frac{1}{D}\}$ theo các bước của thuật toán 3.2 như sau:

Bước 1: $S \leftarrow R_U = \{b, a, e, c\}$;

Bước 2: Áp dụng công thức gia tăng (3.9) ta có:

- $F_s = FIG(D|S, U \cup \Delta_u) = \frac{26}{49} + \frac{4.6}{49}$;

- $F_c = FIG(D|C, U \cup \Delta_u) = \frac{26}{49} + \frac{4.6}{49}$;

Vì $F_s = F_c$, kết thúc thuật toán ta thu được tập rút gọn $R_{U \cup u_7} = \{b, a, e, c\}$.

3.4. Xây dựng thuật toán cập nhật tập rút gọn khi loại bỏ tập đối tượng

Phần này trình bày về phương pháp xây dựng thuật toán cập nhật tập rút gọn khi bảng quyết định NDT loại bỏ tập đối tượng. Các bước chính gồm có: 1) xây dựng công thức tính toán gia tăng khi loại bỏ một đối tượng, 2) xây dựng công thức tính toán gia tăng khi loại bỏ tập đối tượng, 3) xây dựng thuật toán cập nhật tập rút gọn khi loại bỏ tập đối tượng FIGDMO. Trước tiên là phần xây dựng công thức tính toán gia tăng khi loại bỏ một đối tượng cho bảng quyết định NDT.

3.4.1. Công thức gia tăng khi loại bỏ một đối tượng

Mệnh đề 3.7 (Hạt thông tin mờ gia tăng khi loại bỏ một đối tượng)**.** *Cho bảng quyết định NDT* $=< U, C, D >$ *và tập thuộc tính* $P \subseteq C$*, u là đối tượng cần được loại bỏ khỏi bảng quyết định NDT. Khi đó hạt thông tin mờ của P trên* $U - \{u\}$ *được xác định như sau:*

$$FIG(P, U - \{u\}) = 1 - \frac{1}{|U - \{u\}|^2} \left(|U|^2 - |U|^2 FIG(P, U) - 2 \sum_{u=1}^{|U|} \left| [i]_P^{\{u\}} \right| - 1 \right) \tag{3.11}$$

Chứng minh. Dựa theo công thức (3.4), ta có

$$FIG(P, U - \{u\}) = 1 - \frac{\sum_{i=1}^{|U-\{u\}|} \left| [i]_P^{U-\{u\}} \right|}{|U-\{u\}|^2} = 1 - \frac{\sum_{i=1}^{|U|} \left| [i]_P^{U} \right| - \sum_{i=1}^{|U-\{u\}|} \left| [i]_P^{\{u\}} \right| - \sum_{i=1}^{|\{u\}|} [i]_P^{U-\{u\}} - \sum_{i=1}^{|\{u\}|} \left| [i]_P^{\{u\}} \right|}{|U-\{u\}|^2}.$$

Công thức (3.1) có tính chất đối xứng, do đó: $\sum_{i=1}^{|U-\{u\}|} \left| [i]_P^{\{u\}} \right| = \sum_{i=1}^{|\{u\}|} [i]_P^{U-\{u\}}$

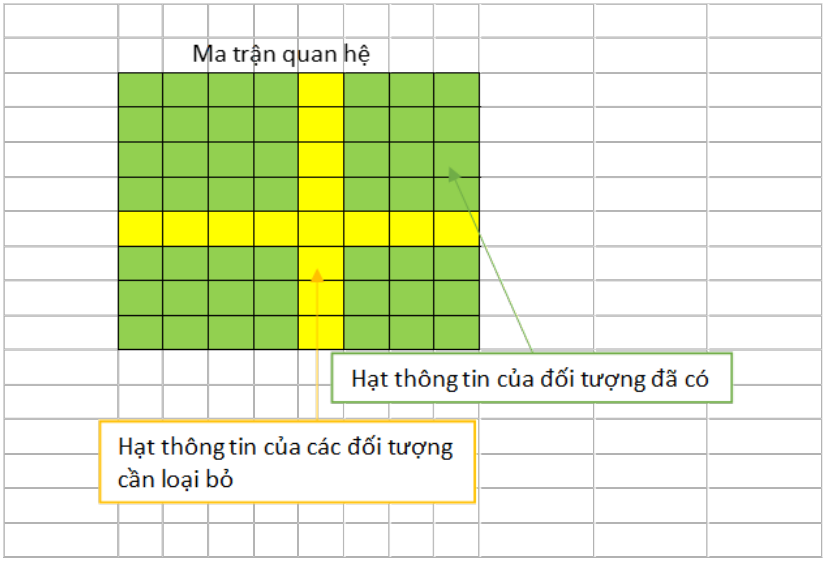
Khi đó:

$$FIG(P,U\cup\{u\}) = 1 - \frac{\sum_{i=1}^{|U|}\left|[i]_P^U\right| - 2\sum_{i=1}^{|U-\{u\}|}\left|[i]_P^{\{u\}}\right| - 1}{|U-\{u\}|^2} = 1 - \frac{|U|^2 - |U|^2 FIG(P,U) - 2\sum_{i=1}^{|U-\{u\}|}\left|[i]_P^{\{u\}}\right| - 1}{|U-\{u\}|^2}$$

$$= 1 - \frac{1}{|U-\{u\}|^2}\left(|U|^2 - |U|^2 FIG(P,U) - 2\sum_{i=1}^{|U-\{u\}|}\left|[i]_P^{\{u\}}\right| - 1\right)$$

Ta có điều phải chứng minh. □

Một cách khái quát, phương pháp tính toán gia tăng khi loại bỏ một đối tượng có thể được minh họa như trong Hình 3.2



Hình 3.2: *Minh họa phương pháp gia tăng khi loại bỏ một đối tượng*

3.4.2. Công thức gia tăng khi loại bỏ một tập đối tượng

Dựa trên công thức gia tăng khi loại bỏ một đối tượng khỏi bảng quyết định NDT, công thức gia tăng được mở rộng cho trường hợp loại bỏ nhiều đối tượng. Trên cơ sở đó, độ đo độ phụ thuộc của thuộc tính sẽ được đề xuất.

Mệnh đề 3.8 (Hạt thông tin mờ gia tăng khi loại bỏ tập đối tượng)**.** *Cho bảng quyết định NDT* $=< U,C,D >$ *và tập thuộc tính* $P \subseteq Q \subseteq C$, $\Delta_u = \{u_1,u_2,...u_k\}$ *là tập đối tượng cần được loại bỏ khỏi NDT. Khi đó hạt thông tin mờ của* P *trên* $U - \Delta_u$ *được*

xác định như sau:

$$FIG(P,U-\Delta_u)=1-\frac{1}{|U-\Delta_u|^2}\left(|U|^2-|U|^2FIG(P,U)-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{\Delta_u}\right|\right) \tag{3.12}$$

Chứng minh. Dựa theo công thức (3.4), ta có $FIG(P,U-\Delta_u)=1-\frac{\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{U-\Delta_u}\right|}{|U-\Delta_u|^2}=1-\frac{\sum_{i=1}^{|U|}\left|[i]_P^{U}\right|-\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{U-\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{\Delta_u}\right|}{|U-\Delta_u|^2}$.

Công thức (3.1) có tính chất đối xứng, do đó: $\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|=\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{U-\Delta_u}\right|$.

Khi đó: $FIG(P,U-\Delta_u)=1-\frac{\sum_{i=1}^{|U|}\left|[i]_P^{U}\right|-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{\Delta_u}\right|}{|U-\Delta_u|^2}$

$=1-\frac{|U|^2-|U|^2FIG(P,U)-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{\Delta_u}\right|}{|U-\Delta_u|^2}$

$=1-\frac{1}{|U-\Delta_u|^2}\left(|U|^2-|U|^2FIG(P,U)-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]_P^{\Delta_u}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]_P^{\Delta_u}\right|\right)$.

Ta có điều phải chứng minh. □

Mệnh đề 3.9 (Hạt thông tin mờ phụ thuộc gia tăng khi loại bỏ tập đối tượng)**.** *Cho bảng quyết định* $NDT=<U,C,D>$ *và tập thuộc tính* $P\subseteq Q\subseteq C$, $\Delta_u=\{u_1,u_2,...u_k\}$ *là tập đối tượng cần được loại bỏ khỏi NDT. Khi đó hạt thông tin mờ của D phụ thuộc vào P trên* $U-\Delta_u$ *được xác định như sau:*

$$FIG(D|P,U-\Delta_u)=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D|P,U)\\+2\sum_{i=1}^{|U-\Delta_u|}\left(\left|[i]_{D\cup P}^{\Delta_u}\right|-\left|[i]_P^{\Delta_u}\right|\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]_{D\cup P}^{\Delta_u}\right|-\left|[i]_P^{\Delta_u}\right|\right)\end{array}\right) \tag{3.13}$$

Chứng minh. Dựa theo định nghĩa 3.5 và mệnh đề 3.4, ta có

$$
\begin{aligned}
&FIG(D|P,U-\Delta_u) = FIG(D\cup P,U-\Delta_u) - FIG(P,U-\Delta_u)\\
&= 1-\frac{1}{|U-\Delta_u|^2}\left(|U|^2-|U|^2FIG(D\cup P,U)-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]^{\Delta_u}_{D\cup P}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]^{\Delta_u}_{D\cup P}\right|\right)\\
&-\left(1-\frac{1}{|U-\Delta_u|^2}\left(|U|^2-|U|^2FIG(P,U)-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]^{\Delta_u}_{P}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]^{\Delta_u}_{P}\right|\right)\right)\\
&=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D\cup P,U)-|U|^2FIG(P,U)\\+\left(2\sum_{i=1}^{|U-\Delta_u|}\left|[i]^{\Delta_u}_{D\cup P}\right|+\sum_{i=1}^{|\Delta_u|}\left|[i]^{\Delta_u}_{D\cup P}\right|-2\sum_{i=1}^{|U-\Delta_u|}\left|[i]^{\Delta_u}_{P}\right|-\sum_{i=1}^{|\Delta_u|}\left|[i]^{\Delta_u}_{P}\right|\right)\end{array}\right)\\
&=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}|U|^2FIG(D|P,U)\\+2\sum_{i=1}^{|U-\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup P}\right|-\left|[i]^{\Delta_u}_{P}\right|\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup P}\right|-\left|[i]^{\Delta_u}_{P}\right|\right)\end{array}\right)
\end{aligned}
$$

Ta có điều phải chứng minh. □

3.4.3. Đề xuất thuật toán gia tăng khi loại bỏ tập đối tượng

Trước khi đề xuất thuật toán, độ đo độ quan trọng của thuộc tính được mở rộng thông qua mệnh đề sau đây:

Mệnh đề 3.10 (Độ quan trọng của thuộc tính gia tăng khi loại bỏ một tập đối tượng)**.** *Cho bảng quyết định $NDT =< U,C,D >$ và $B \subseteq C$, độ quan trọng của thuộc tính $a \in C-B$ với B trên $U-\Delta_u$ được xác định như sau:*

$$
Sig^{U-\Delta_u}_{D|B}(a) = \frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}|U|^2Sig^{U}_{D|B}(a)\\+2\sum_{i=1}^{|U-\Delta_u|}\left(\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\right)\\+\sum_{i=1}^{|\Delta_u|}\left(\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\right)\end{array}\right) \tag{3.14}
$$

Chứng minh. Dựa theo định nghĩa 3.5 và mệnh đề 3.4, ta có

$$
\begin{aligned}
&FIG(D|B\cup\{a\},U-\Delta_u)-FIG(D|B,U-\Delta_u)\\
&=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}
|U|^2FIG(D|B\cup\{a\},U)\\
-2\sum_{i=1}^{|U-\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\\
-\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|-\left|[u]^{\Delta_u}_{B\cup\{a\}}\right|\right)
\end{array}\right)\\
&-\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}
|U|^2FIG(D|B,U)-2\sum_{i=1}^{|U-\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{B}\right|\right)\\
-\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{B}\right|\right)
\end{array}\right)\\
&=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}
|U|^2FIG(D|B\cup\{a\},U)-|U|^2FIG(D|B,U)\\
+2\sum_{i=1}^{|U-\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|+\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\\
+\sum_{i=1}^{|\Delta_u|}\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|+\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)
\end{array}\right)\\
&=\frac{1}{|U-\Delta_u|^2}\left(\begin{array}{l}
|U|^2Sig^U_{D|B}(a)\\
+2\sum_{i=1}^{|U-\Delta_u|}\left(\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\right)\\
+\sum_{i=1}^{|\Delta_u|}\left(\left(\left|[i]^{\Delta_u}_{D\cup B}\right|-\left|[i]^{\Delta_u}_{D\cup B\cup\{a\}}\right|\right)+\left(\left|[i]^{\Delta_u}_{B}\right|-\left|[i]^{\Delta_u}_{B\cup\{a\}}\right|\right)\right)
\end{array}\right)
\end{aligned}
$$

Ta có điều phải chứng minh. □

Dựa trên mệnh đề 3.10. Thuật toán FIGDMO cập nhật tập rút gọn khi loại bỏ tập đối tượng được đề xuất, sau đây là các bước chính của thuật toán FIGAMO. Bước 2, khởi tạo $S = R_U$, với R_U là tập rút gọn thu được trên bảng quyết định NDT với U đối tượng ban đầu. Bước 3, dựa theo mệnh đề 3.9, các biến F_s và F_c lưu các giá trị tương ứng của $FIG(D|S,U-\Delta_u)$ và $FIG(D|C,U-\Delta_u)$. Bước 4, nếu độ chênh lệch của F_s và F_c nhỏ hơn 0.5% thì trả lại S, nghĩa là tập rút gọn không thay đổi. Ngược lại, thực hiện các bước trong vòng lặp từ bước 8 đến 12 để cập nhật lại tập rút gọn S. Vòng lặp các bước từ 15 đến 17 nhằm loại bỏ các thuộc tính dư thừa nếu có. Bước 20 trả lại tập rút gọn được cập nhật của bảng quyết định NDT khi bổ sung Δ_u thuộc tính.

Sau đây là phần đánh giá độ phức tạp của thuật toán FIFAMO. Bước 2 có độ phức tạp xấp xỉ là $O\left(|\Delta_u|^2|C|\right)$. Dựa trên mệnh đề 3.6, độ phức tạp các bước từ 6 đến 12 là

Thuật toán 3.3 Thuật toán rút gọn thuộc tính gia tăng khi loại bỏ tập đối tượng theo tiếp cận hạt thông tin mờ FIGAMO

Input: $NDT =< U, C, D >$, R_U, Δ_u
Output: Tập rút gọn $R_{U-\Delta_u}$

1: $S \leftarrow R_U$;
2: $F_s = FIG(D|S, U-\Delta_u), F_c = FIG(D|C, U-\Delta_u)$;
3: **if** $|F_c - F_s| \leq 0.5\%$ **then**
4: **return** S;
5: **end if**
6: **while** $|F_c - F_s| > 0.5\%$ **do**
7: **for all** $a \in (C-S)$ **do**
8: **compute** $Sig_{D|S}^{U-\Delta_u}(a)$
9: **end for**
10: $s = max\left\{Sig_{D|S}^{U-\Delta_u}(a)\right\}$
11: $S \leftarrow S \cup \{s\}$
12: **end while**
13: **for all** $a \in S$ **do**
14: $F_{s'} = FIG(D|S-\{a\}, U-\Delta_u)$;
15: **if** $|F_c - F_{s'}| \leq 0.5\%$ **then**
16: $S \leftarrow S-\{a\}$
17: **end if**
18: **end for**
19: $R_{U-\Delta_u} \leftarrow S$
20: **return** $R_{U \cup \Delta_u}$

$O\left(|\Delta_u|^2 |C-S|^2\right)$. Các bước từ 13 đến 18 có độ phức tạp xấp xỉ là $O\left(|\Delta_u|^2 |S|\right)$. Do đó, độ phức tạp của thuật toán FIGDMO là $\max\left\{O\left(|\Delta_u|^2 |C-S|^2\right), O\left(|\Delta_u|^2 |C|\right)\right\}$, tương đương với thuật toán FDMDO [76].

Tuy nhiên, cũng giống như thuật toán FIG, thuật toán FIGDMO có thời gian thực hiện trong thực tế nhanh hơn thuật toán FDMDO do kích thước tập rút gọn được giới hạn theo độ lệch của hạt là 0.5% so với tập thuộc tính gốc.

3.5. Thực nghiệm và đánh giá các thuật toán

Để chứng minh các thuật toán rút gọn thuộc tính gia tăng FIGAMO và FIGDMO hiệu quả về thời gian rút gọn thuộc tính, phần này trình bày kết quả thực nghiệm của các thuật toán FIG và FIGAMO. Tiến hành so sánh với các thuật toán được xây dựng theo tiếp cận FD. Nội dung này chỉ thực nghiệm các thuật toánFIGAMO để đánh giá hiệu năng của tiếp cận tính toán gia tăng đề xuất. Hơn nữa quá việc tính toán gia tăng khi bổ sung tập đối tượng thường được sử dụng để cải tiến thời gian thực hiện của các thuật toán rút gọn theo tiếp cận filter truyền thống.

3.5.1. Kịch bản và môi trường thực nghiệm

Để đánh giá tính hiệu quả về thời gian rút gọn thuộc tính của các thuật toán đề xuất, so sánh độ chính xác phân lớp và kích thước của tập rút gọn thu được từ các thuật toán. Kịch bản thực nghiệm của các thuật toán đề xuất được thực hiện như sau

- Thứ nhất, đánh giá thuật toán FIG. Về cơ bản độ phức tạp của các công thức tính toán gia tăng của hai tiếp cận FIG và FD là như nhau. Tuy nhiên thuật toán rút gọn thuộc tính theo tiếp cận FIG được thiết kế với điều kiện dừng là 0.5% về độ lệch hạt thông tin. Qua đó, ta có thể xác minh tính đúng đắn trong phần đánh giá độ phức tạp của thuật toán FIG

- Thứ hai, đánh giá thuật toán FIGAMO. Phần đánh giá này cũng nhằm xác minh tính hiệu quả về thời gian của thuật toán đề xuất so với thuật toán FDMAO. Qua đó có thể xác minh điều kiện dừng của thuật toán đề xuất ảnh hưởng thế nào tới thời gian rút gọn thuộc tính gia tăng khi bổ sung tập đối tượng.

Các tập dữ liệu thực nghiệm được tải về từ kho dữ liệu chuẩn của UCI (UC-Irvine Machine Learning Repository) [93]. Chi tiết các bộ dữ liệu có thể tìm thấy tại Bảng 3.1, trong đó $|U|$ là số đối tượng, $|C|$ là số các thuộc tính điều kiện, và $|D|$ là số phân lớp của tập dữ liệu. Các thuộc tính điều kiện của các tập dữ liệu này đều có miền giá trị số liên tục. Trước khi thực hiện rút gọn thuộc tính với các thuật toán, dữ liệu được

Bảng 3.1: *Bảng mô tả dữ liệu gia tăng*

STT	Tập dữ liệu	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	BCWD	Breast Cancer Wisconsin (Diagnostic)	569	30	2
5	IS	Ionosphere	351	34	2
6	SHDC	SPECTF Heart Data Set	267	44	2

chuẩn hóa về đoạn $[0,1]$.

Tập rút gọn thu được từ các thuật toán được kiểm thử trên hai mô hình phân lớp dữ liệu số là SVM và k-NN với ($k = |D|$). Phương pháp đánh giá chéo 10-folde kết hợp với các độ đo (accuarcy, precision, recall) để đánh giá độ chính xác trung bình mô hình phân lớp xây dựng trên tập dữ liệu rút gọn. Ngôn ngữ lập trình Python được sử dụng để cài đặt các thuật toán trên nền tảng Windows 10 và phần cứng với bộ xử lý i5, 8GB RAM.

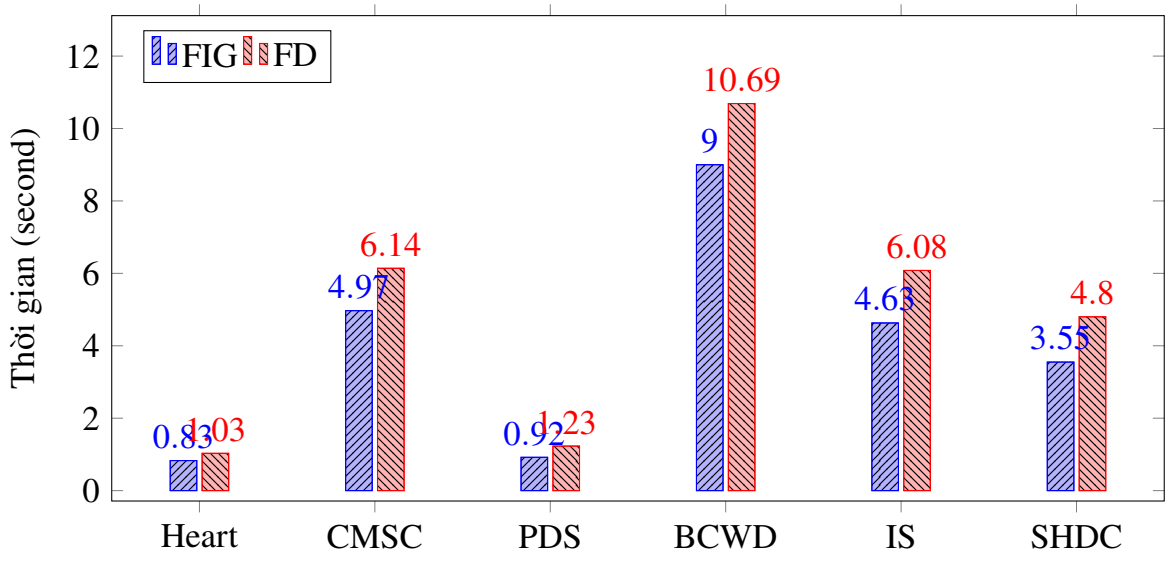
3.5.2. Đánh giá thuật toán FIG

Phần này sử dụng thuật toán FD [76] để so sánh với thuật toán đề xuất FIG. Đây là thuật toán được xây dựng dựa trên độ đo khoảng cách mờ, rút gọn thuộc tính cho bảng quyết định tĩnh. Trước tiên là phần so sánh về thời gian thực hiện của các thuật toán.

3.5.2.1. Thời gian thực hiện

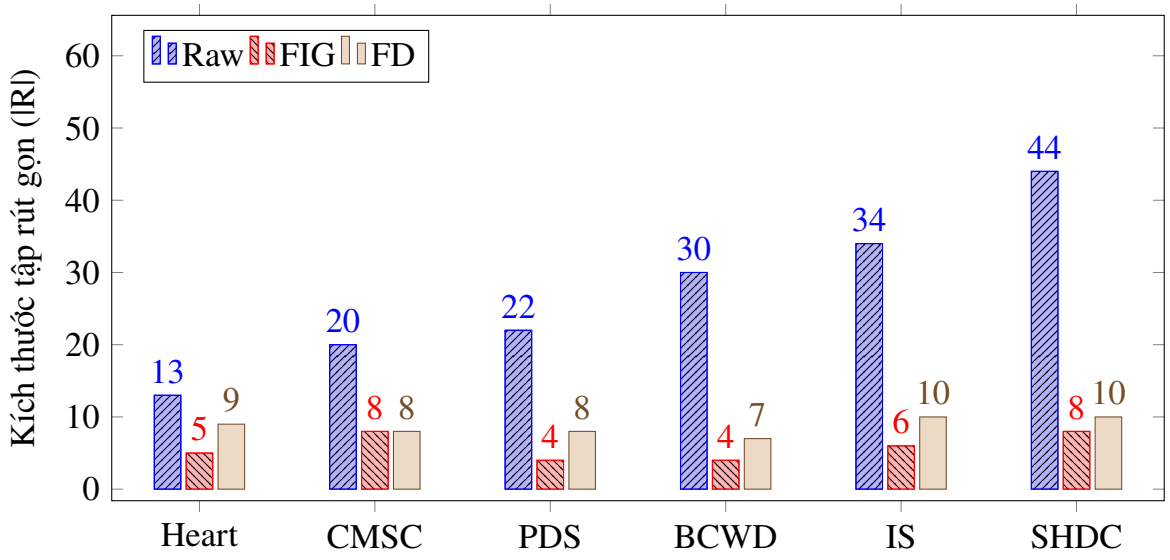
Hình 3.3 so sánh thời gian thực hiện của các thuật toán. Chúng ta có thể thấy thời gian rút gọn thuộc tính của thuật toán đề xuất FIG tốt hơn thuật toán FD trên mọi bộ dữ liệu thử nghiệm. Điều đó càng khẳng định, điều kiện dừng được thiết kế trong thuật toán đề xuất ảnh hưởng tốt tới thời gian thực hiện của thuật toán FIG. Trong quá trình thực nghiệm tác giả nhận thấy, càng tăng độ lệch của điều kiện dừng, thuật toán

thực hiện càng nhanh và ngược lại. Khi độ lệch bằng không, thời gian thực hiện của thuật toán FIG xấp xỉ thời gian thực hiện của thuật toán FD.



Hình 3.3: *So sánh thời gian thực hiện giữa các thuật toán rút gọn thuộc tính*

3.5.2.2. Kích thước tập rút gọn



Hình 3.4: *So sánh kích thước tập rút gọn giữa các thuật toán rút gọn thuộc tính*

Hình 3.4 so sánh kích thước tập rút gọn thu được của các thuật toán. Chúng ta có thể thấy kích thước tập rút gọn của thuật toán đề xuất FIG tốt hơn thuật toán FD trên

mọi bộ dữ liệu thử nghiệm. Điều đó càng khẳng định, điều kiện dừng được thiết kế trong thuật toán đề xuất ảnh hưởng tốt tới khả năng giảm thuộc tính của thuật toán FIG. Trong quá trình thực nghiệm tác giả cũng nhận thấy, càng tăng độ lệch của điều kiện dừng, kích thước tập rút gọn càng nhỏ và ngược lại. Khi độ lệch bằng không, kích thước tập rút gọn thu được từ thuật toán FIG không chênh lệch đáng kể với tập rút gọn của thuật toán FD. Như vậy ta có thể thấy mối quan hệ tuyến tính giữa thời gian thực hiện và kích thước tập rút gọn của thuật toán đề xuất là tuyến tính. Thời gian thực hiện càng nhanh, tập rút gọn thu được càng nhỏ và ngược lại.

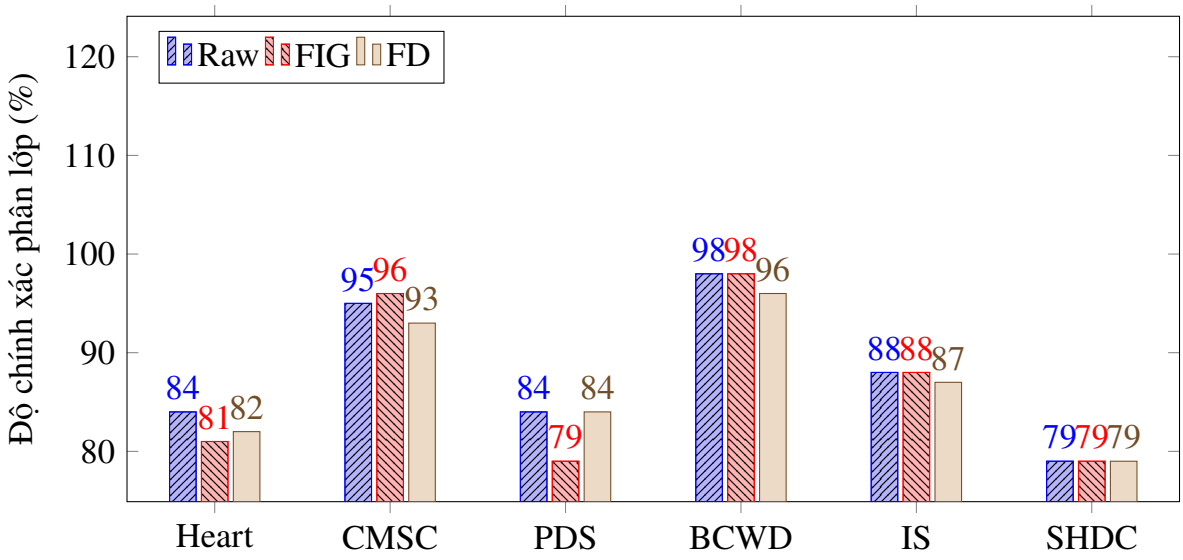
3.5.2.3. Độ chính xác phân lớp

Hình 3.5 so sánh độ chính xác phân lớp của các tập thuộc tính rút gọn, sử dụng mô hình SVM. Kết quả cho thấy, trên các tập dữ liệu có số lượng mẫu nhỏ (dưới 500 mẫu), độ chính xác phân lớp của tập rút gọn không khác biệt đáng kể so với tập thuộc tính gốc. Điều này cho thấy thuật toán rút gọn hoạt động hiệu quả trong việc duy trì khả năng phân loại khi dữ liệu hạn chế.

Tuy nhiên, đối với các tập dữ liệu lớn hơn (trên 500 mẫu) như CMSC và BCWD, tập thuộc tính rút gọn thu được từ thuật toán FIG (được đề xuất trong luận án) thể hiện ưu thế vượt trội so với thuật toán FD (một thuật toán đối sánh). Điều này cho thấy thuật toán FIG có khả năng xử lý tốt hơn với dữ liệu phức tạp, mang lại kết quả phân loại chính xác hơn khi số lượng mẫu lớn.

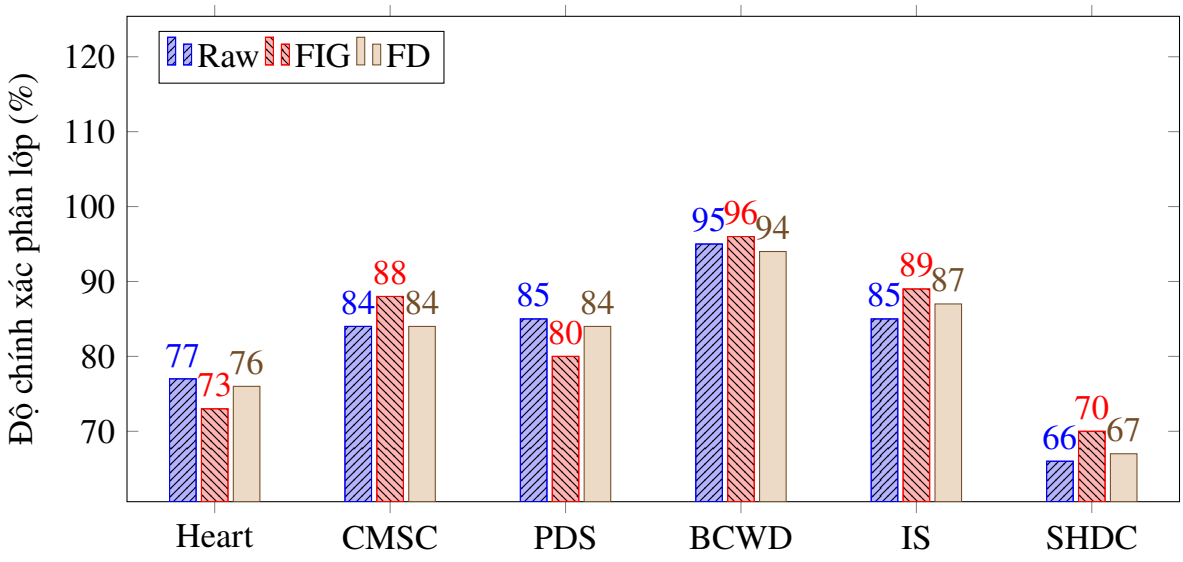
Tương tự như kết quả trên mô hình SVM, Hình 3.6 thể hiện xu hướng tương tự trên mô hình phân lớp k-NN. Cụ thể, đối với hai bộ dữ liệu CMSC và BCWD, tập thuộc tính rút gọn do thuật toán FIG tạo ra có độ chính xác phân loại vượt trội so với thuật toán FD. Điều này củng cố thêm bằng chứng về khả năng của thuật toán FIG trong việc xử lý các tập dữ liệu có số mẫu cao.

Ngược lại, trên các bộ dữ liệu còn lại, sự khác biệt về độ chính xác giữa tập rút gọn từ thuật toán FIG và thuật toán FD là không đáng kể. Điều này có thể do đặc điểm của các bộ dữ liệu này ít phức tạp hơn, khiến sự khác biệt giữa hai thuật toán trở nên mờ



Hình 3.5: *So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình SVM*

nhạt hơn. Tuy nhiên, kết quả trên CMSC và BCWD vẫn chứng minh được tiềm năng của FIG trong các tình huống đòi hỏi khả năng rút gọn thuộc tính trên các bộ dữ liệu có số mẫu lớn.



Hình 3.6: *So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình KNN*

Kết quả thực nghiệm cho thấy thuật toán FIG, được đề xuất trong nghiên cứu này, mang lại nhiều ưu điểm vượt trội trong quá trình rút gọn thuộc tính. Không chỉ sở hữu thời gian rút gọn hiệu quả và tạo ra các tập thuộc tính rút gọn có kích thước nhỏ hơn

trên toàn bộ các tập dữ liệu thử nghiệm, FIG còn đạt được độ chính xác phân lớp cao hơn trên một số bộ dữ liệu lớn.

Những ưu điểm này nhấn mạnh tính hiệu quả của việc sử dụng độ đo hạt thông tin trên không gian xấp xỉ mờ trong quá trình lựa chọn thuộc tính. Khả năng này giúp FIG xác định và loại bỏ các thuộc tính không liên quan hoặc dư thừa, đồng thời giữ lại những thuộc tính quan trọng nhất để đảm bảo độ chính xác của mô hình phân loại. Kết quả này củng cố thêm giá trị của phương pháp tiếp cận dựa trên không gian xấp xỉ mờ trong bài toán rút gọn thuộc tính.

3.5.3. Đánh giá thuật toán FIGAMO

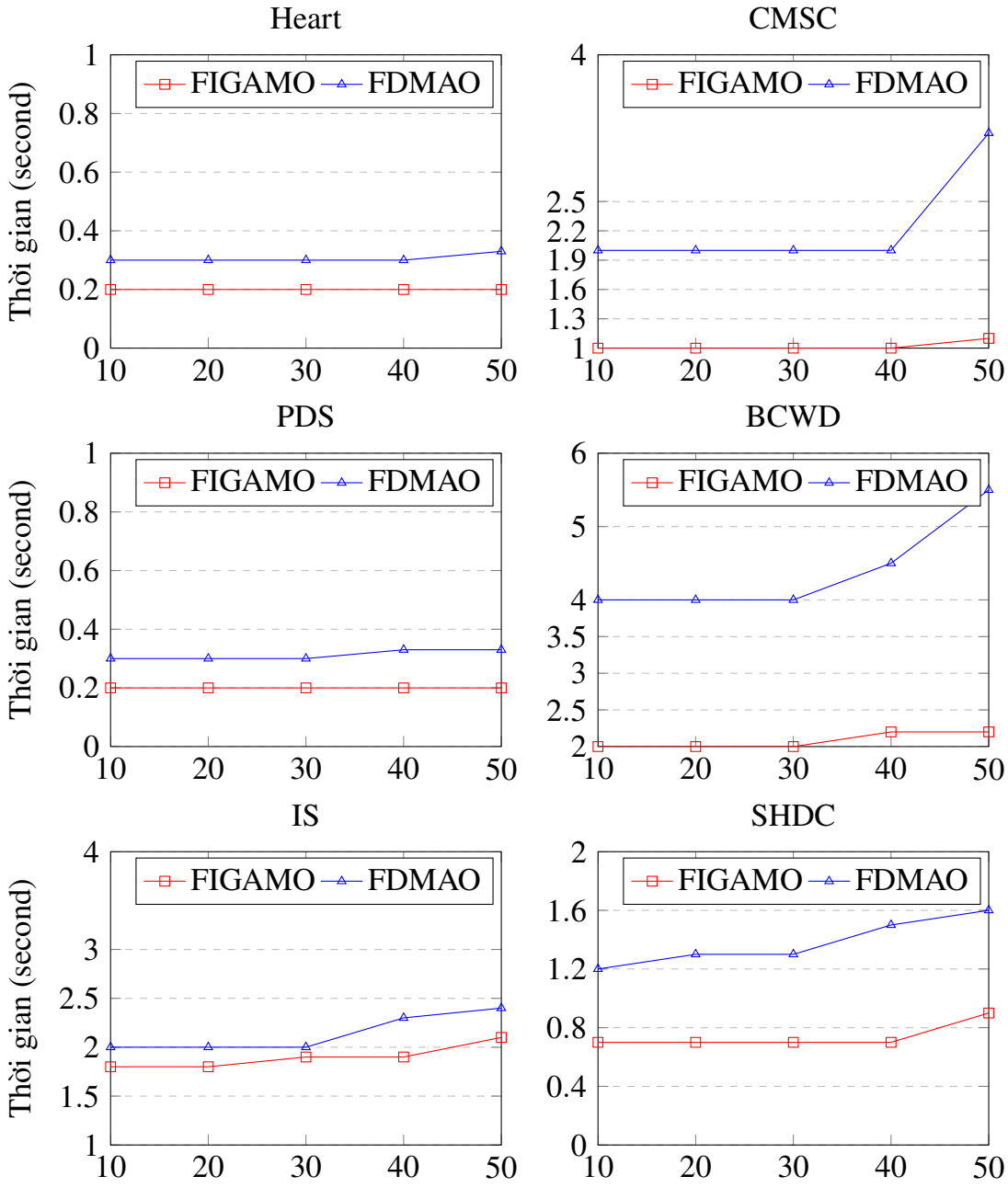
Phần này sử dụng thuật toán FDMAO [76] để so sánh với thuật toán đề xuất FIGAMO. Đây là thuật toán gia tăng được xây dựng dựa trên độ đo khoảng cách mờ, rút gọn thuộc tính cho bảng quyết định khi bổ sung tập đối tượng.

Luận án chỉ thực nghiệm thuật toán FIGAMO mà không thực nghiệm thuật toán FIGDMO để đánh giá thời gian rút gọn thuộc tính. Lý do, về mặt lý thuyết hai thuật toán đều có độ phức tạp là như nhau khi cùng số lượng đối tượng bổ sung hay loại bỏ, do phần dữ liệu trước đó đã được tính toán. Tuy nhiên, phương pháp tính toán gia tăng có thể ứng dụng rút gọn thuộc tính cho bảng quyết định tĩnh nhằm cải thiện thời gian rút gọn thuộc tính cho bảng quyết định. Đây cũng là hướng nghiên cứu tiếp theo trong tương lai của luận án.

3.5.3.1. Thời gian thực hiện

Hình 3.7 so sánh thời gian thực hiện của các thuật toán. Chúng ta có thể thấy thời gian rút gọn thuộc tính của thuật toán đề xuất FIGAMO tốt hơn thuật toán FDMAO trên mọi bộ dữ liệu thử nghiệm. Điều đó càng khẳng định, điều kiện dừng được thiết kế trong thuật toán đề xuất ảnh hưởng tốt tới thời gian thực hiện của thuật toán FIGAMO.

Trong quá trình thử nghiệm với các ngưỡng khác nhau tác giả nhận thấy, càng tăng độ lệch của điều kiện dừng, thuật toán thực hiện càng nhanh và ngược lại. Khi độ lệch



Hình 3.7: *So sánh thời gian thực hiện giữa các thuật toán khi bổ sung tập đối tượng*

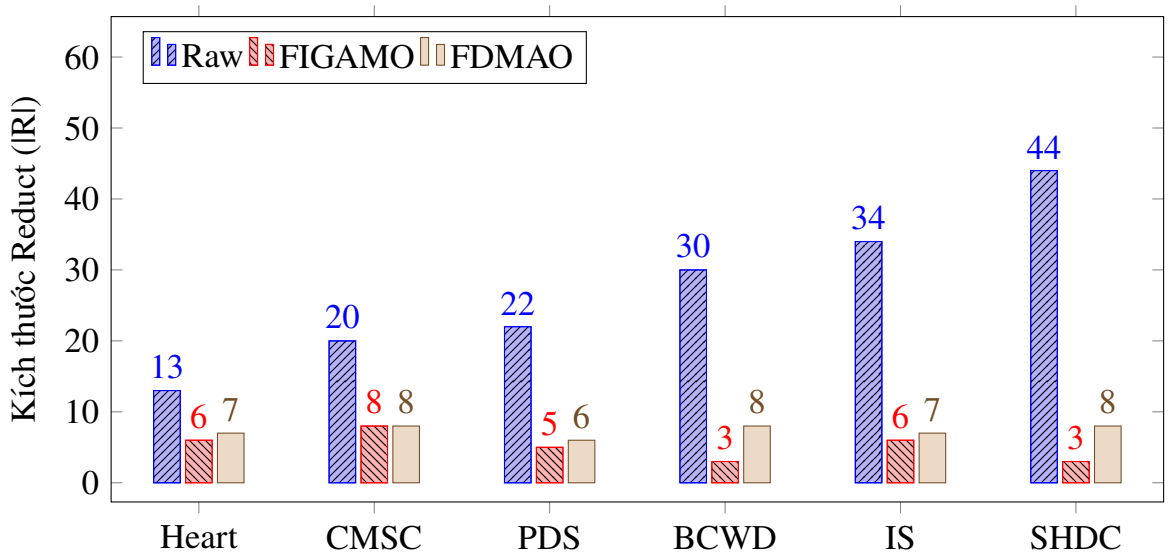
bằng không, thời gian thực hiện của thuật toán FIGAMO xấp xỉ thời gian thực hiện của thuật toán FDMAO. Để cân bằng độ chính xác cũng như thời gian tính toán tập rút gọn của thuật toán đề xuất, tác giả lựa chọn giá trị độ lệch là 0.5% căn cứ trên kích thước (số mẫu) nhỏ nhất và kích thước lớn nhất của các tập dữ liệu được thực nghiệm.

Mức độ chênh lệch về thời gian thực hiện giữa các mức tính toán gia tăng khác

nhau của hai thuật toán là rất thấp trên hầu hết các bộ dữ liệu. Tuy nhiên, quan sát bộ dữ liệu CSMC và BCWD cho thấy thời gian thực hiện của thuật toán FDMAO tăng đột ngột ở tại mức bổ sung 50% bộ dữ liệu. Trong khi đó thuật toán đề xuất không bị ảnh hưởng nhiều. Nguyên nhân là do ngưỡng 0.5% được áp dụng cho toàn bộ mức tính toan gia tăng của thuật toán đề xuất.

3.5.3.2. Kích thước tập rút gọn

Tiếp theo là phần đánh giá về kích thước và độ chính xác phân lớp trên hai mô hình phân lớp SVM và k-NN, tập rút gọn dùng để đánh giá là tập rút gọn cuối cùng (tại mức bổ sung 50%) của quá trình tính toán gia tăng tập rút gọn. Hình 3.8 so sánh kích



Hình 3.8: *So sánh kích thước tập rút gọn giữa các thuật toán khi bổ sung tập đối tượng*

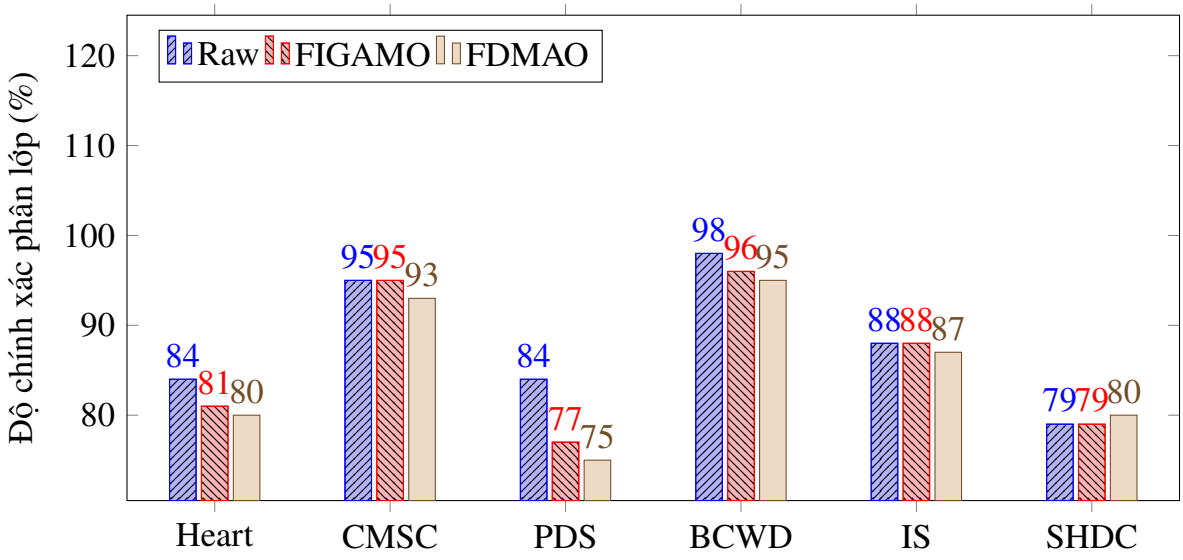
thước tập rút gọn thu được của các thuật toán. Chúng ta có thể thấy kích thước tập rút gọn của thuật toán đề xuất FIGAMO tốt hơn thuật toán FDMAO trên mọi bộ dữ liệu thử nghiệm. Điều đó cho thấy tính bảo toàn về khả năng nén dữ liệu của thuật toán FIG trên thuật toán FIGAMO.

Dựa trên biểu đồ, cả thuật toán FIGAMO và FDMAO đều giảm đáng kể kích thước tập thuộc tính so với tập gốc ("Raw") trên mọi bộ dữ liệu, cho thấy hiệu quả trong

việc loại bỏ thuộc tính dư thừa. Tuy nhiên, FIGAMO thường tạo ra tập rút gọn nhỏ hơn FDMAO, đặc biệt rõ rệt trên BCWD và SHDC, trong khi trên CMSC kích thước tập rút gọn tương đương. Ví dụ, trên Heart, FIGAMO (6) và FDMAO (7) gần bằng nhau, nhưng đều nhỏ hơn Raw (13). Sự khác biệt lớn hơn trên BCWD, với FIGAMO (3) nhỏ hơn nhiều so với FDMAO (8) và Raw (30). Tương tự, trên SHDC, FIGAMO (3) nhỏ hơn đáng kể so với FDMAO (8) và Raw (44). Nhìn chung, FIGAMO có xu hướng rút gọn thuộc tính hiệu quả hơn trong việc giảm số lượng thuộc tính, đặc biệt trên các tập dữ liệu lớn.

3.5.3.3. Độ chính xác phân lớp

Hình 3.9 là biểu đồ so sánh độ chính xác phân lớp của các thuật toán rút gọn thuộc tính (FIGAMO và FDMAO) so với việc sử dụng toàn bộ thuộc tính gốc (Raw) trên mô hình phân lớp SVM.



Hình 3.9: *So sánh độ chính xác phân lớp của các thuật toán trên mô hình SVM khi bổ sung tập đối tượng*

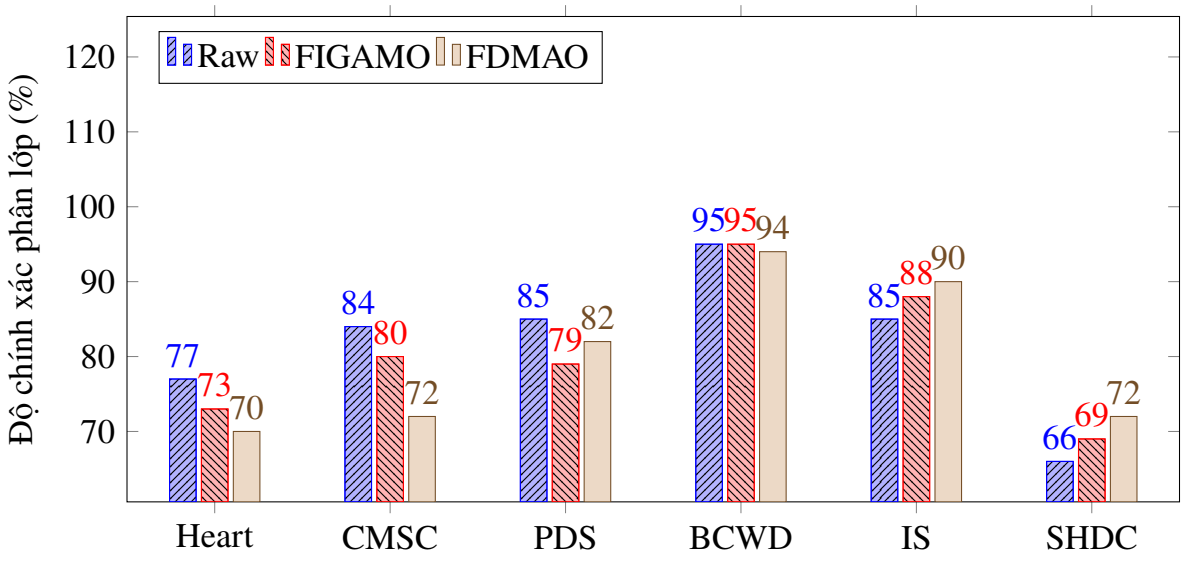
Quan sát biểu đồ cho thấy, trên nhiều tập dữ liệu như Heart, CMSC, BCWD và IS, độ chính xác phân lớp khi sử dụng tập thuộc tính rút gọn bởi FIGAMO rất gần hoặc thậm chí bằng với độ chính xác của tập thuộc tính gốc. Thuật toán FIGAMO thường

đạt độ chính xác cao hơn hoặc tương đương so với FDMAO trên các bộ dữ liệu này.

Tuy nhiên, trên bộ dữ liệu PDS, độ chính xác của cả hai thuật toán rút gọn đều giảm đáng kể so với tập gốc, mặc dù FIGAMO vẫn tốt hơn FDMAO một chút. Ngược lại, trên bộ dữ liệu SHDC, thuật toán FDMAO lại cho độ chính xác cao hơn so với cả FIGAMO và tập thuộc tính gốc.

Nhìn chung, kết quả trên mô hình SVM cho thấy thuật toán FIGAMO có khả năng duy trì độ chính xác phân lớp hiệu quả sau khi rút gọn thuộc tính trên đa số các tập dữ liệu, và thường vượt trội hơn so với FDMAO.

Hình 3.10 so sánh độ chính xác phân lớp của các thuật toán rút gọn thuộc tính (FIGAMO và FDMAO) so với tập thuộc tính gốc (Raw) trên mô hình k-NN.



Hình 3.10: *So sánh độ chính xác phân lớp của các thuật toán trên mô hình KNN khi bổ sung tập đối tượng*

Trên các bộ dữ liệu Heart, CMSC, và PDS, cả FIGAMO và FDMAO đều cho độ chính xác phân lớp thấp hơn so với tập thuộc tính gốc. Tuy nhiên, trên bộ dữ liệu BCWD, FIGAMO đạt độ chính xác gần tương đương với Raw. Đáng chú ý, trên IS và SHDC, cả FIGAMO và FDMAO đều cho độ chính xác cao hơn đáng kể so với tập thuộc tính gốc.

So sánh FIGAMO và FDMAO, ta thấy rằng FIGAMO thường cho độ chính xác

cao hơn FDMAO trên hầu hết các bộ dữ liệu, ngoại trừ IS, nơi FDMAO vượt trội hơn.

Nhìn chung, kết quả trên mô hình k-NN cho thấy hiệu quả của việc rút gọn thuộc tính phụ thuộc nhiều vào từng bộ dữ liệu cụ thể. Trong một số trường hợp, việc rút gọn có thể làm giảm độ chính xác, nhưng trong những trường hợp khác, nó có thể cải thiện đáng kể độ chính xác phân lớp. FIGAMO có vẻ là một lựa chọn tốt hơn FDMAO trên đa số các bộ dữ liệu.

3.6. Kết luận Chương 3

Chương 3 của luận án tập trung vào việc phát triển một phương pháp rút gọn thuộc tính gia tăng hiệu quả, sử dụng tiếp cận tính toán hạt trên không gian xấp xỉ mờ. Phương pháp này được thiết kế để xử lý các bộ dữ liệu lớn một cách hiệu quả, đặc biệt khi dữ liệu liên tục được cập nhật hoặc thay đổi.

Điểm nổi bật của phương pháp là việc mở rộng độ đo hạt thông tin trên không gian xấp xỉ mờ. Bằng cách tận dụng khái niệm hạt thông tin, công thức tính toán được đơn giản hóa, giúp giảm thiểu đáng kể thời gian cần thiết để tính toán gia tăng và cập nhật tập rút gọn. Điều này đặc biệt quan trọng khi xử lý dữ liệu lớn, nơi mà thời gian tính toán trở thành một yếu tố then chốt.

Kết quả thực nghiệm cho thấy thuật toán đề xuất không chỉ có thời gian tính toán hiệu quả hơn so với các phương pháp khác, mà còn tạo ra các tập thuộc tính rút gọn có kích thước nhỏ hơn trên mọi bộ dữ liệu thử nghiệm. Quan trọng hơn, thuật toán còn cho thấy sự cải thiện về độ chính xác phân lớp trên một số bộ dữ liệu có số lượng mẫu lớn.

Mặc dù thuật toán đề xuất hứa hẹn trong việc rút gọn thuộc tính cho dữ liệu lớn, nghiên cứu cũng chỉ ra rằng cần phải khảo sát và điều chỉnh giá trị ngưỡng điều chỉnh về độ lệch trước khi áp dụng. Việc này nhằm đảm bảo sự cân bằng giữa thời gian tính toán gia tăng tập rút gọn và độ chính xác phân lớp của tập rút gọn cuối cùng. Sự cân bằng này là rất quan trọng để đạt được hiệu quả tốt nhất trong các ứng dụng thực tế.

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 chọn lọc thuộc tính cho bảng quyết định miền giá trị số. Trước tiên luận án tập trung khảo sát các khoảng trống nghiên cứu 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 chọn lọc thuộc tính cho bảng quyết định theo tiếp cận mở rộng tập thô mờ độ chính xác thay đổi trên không gian xấp xỉ mờ.*

Điểm cốt lõi của phương pháp nằm ở việc xây dựng một độ đo để đánh giá mức độ quan trọng của từng thuộc tính. Độ đo này dựa trên độ phụ thuộc, được định nghĩa theo cách tiếp cận VPOFRS, cho phép xác định chính xác mức độ ảnh hưởng của mỗi thuộc tính đến quá trình phân loại dữ liệu. Điều này giúp chọn ra những thuộc tính quan trọng nhất, đồng thời loại bỏ những thuộc tính ít liên quan hoặc gây nhiễu.

Để kiểm tra hiệu quả của phương pháp đề xuất, luận án đã tiến hành các thử nghiệm trên nhiều tập dữ liệu khác nhau. Kết quả cho thấy rằng thuật toán rút gọn thuộc tính mới không chỉ cải thiện đáng kể thời gian tính toán mà còn tạo ra các tập rút gọn có độ chính xác cao hơn và kích thước nhỏ hơn so với các thuật toán truyền thống. Điều này cho thấy phương pháp mới không chỉ nhanh hơn mà còn hiệu quả hơn trong việc đơn giản hóa mô hình và cải thiện khả năng tổng quát hóa của mô hình.

Ngoài việc áp dụng cho bài toán rút gọn thuộc tính, luận án cũng chỉ ra rằng mô hình tập thô VPOFRS có tiềm năng ứng dụng rộng rãi trong các bài toán phân tích dữ liệu khác. Với khả năng xử lý dữ liệu mờ và không chắc chắn, VPOFRS có thể trở thành một công cụ mạnh mẽ trong nhiều lĩnh vực khác nhau, từ nhận dạng mẫu đến dự báo và ra quyết định, mở ra nhiều hướng nghiên cứu và ứng dụng trong tương lai.

Thứ hai, luận án đề xuất *phương pháp tính toán gia tăng tập rút gọn sử dụng phương pháp tính toán hạt thông tin trên không gian xấp xỉ mờ.*

Điểm nổi bật của phương pháp là việc mở rộng độ đo hạt thông tin trên không gian xấp xỉ mờ. Bằng cách tận dụng khái niệm hạt thông tin, công thức tính toán được đơn giản hóa, giúp giảm thiểu đáng kể thời gian cần thiết để tính toán gia tăng và cập nhật tập rút gọn. Điều này đặc biệt quan trọng khi xử lý dữ liệu lớn, nơi mà thời gian tính toán trở thành một yếu tố then chốt.

Mặc dù thuật toán đề xuất hứa hẹn trong việc rút gọn thuộc tính cho dữ liệu lớn, nghiên cứu cũng chỉ ra rằng cần phải khảo sát và điều chỉnh giá trị ngưỡng điều chỉnh về độ lệch trước khi áp dụng. Việc này nhằm đảm bảo sự cân bằng giữa thời gian tính toán gia tăng tập rút gọn và độ chính xác phân lớp của tập rút gọn cuối cùng. Sự cân bằng này là rất quan trọng để đạt được hiệu quả tốt nhất trong các ứng dụng thực tế.

B. Hướng phát triển tiếp theo của luận án

Các kết quả nghiên cứu hiện tại của luận án được xây dựng dựa trên không gian xấp xỉ mờ, nơi các bảng quyết định số được sử dụng làm dữ liệu đầu vào. Tuy nhiên, một hạn chế của cách tiếp cận này là trong thực tế, nhiều bài toán phân tích dữ liệu liên quan đến các bảng quyết định phức tạp hơn, không chỉ chứa các thuộc tính số (liên tục) mà còn bao gồm các thuộc tính phân loại xen kẽ. Sự hiện diện của cả hai loại thuộc tính này trong cùng một bảng quyết định tạo ra những thách thức riêng trong quá trình rút gọn thuộc tính.

Do đó, một hướng phát triển tiềm năng cho các kết quả nghiên cứu của luận án là mở rộng sang không gian lân cận. Việc mở rộng này sẽ cho phép áp dụng các phương pháp rút gọn thuộc tính đã phát triển cho các bảng quyết định hỗn hợp, bao gồm cả thuộc tính số và thuộc tính phân loại.

Bằng cách mở rộng sang không gian lân cận, luận án có thể giải quyết được một phạm vi lớn hơn các bài toán thực tế. Điều này sẽ làm tăng tính ứng dụng và giá trị thực tiễn của các kết quả nghiên cứu, vì dữ liệu trong thế giới thực thường không thuần nhất và chứa nhiều loại thuộc tính khác nhau. Việc phát triển các phương pháp

rút gọn thuộc tính hiệu quả cho dữ liệu hỗn hợp sẽ là một đóng góp quan trọng cho lĩnh vực khai phá dữ liệu.

DANH MỤC CÁC CÔNG TRÌNH NGHIÊN CỨU

[CT1] **Phạm Minh Ngọc Hà**, Nguyễn Long Giang, Nguyễn Văn Thiện, Nguyễn Bá Quảng, "Về một thuật toán gia tăng tìm tập rút gọn của bảng quyết định không đầy đủ", *Chuyên san các công trình nghiên cứu phát triển CNTT&TT, Tạp chí Công nghệ thông tin và truyền thông - Bộ TT&TT*, Tập 2019, Số 1, Tháng 9, Tr. 11-18.

[CT2] **Phạm Minh Ngọc Hà**, Nguyễn Long Giang, Trần Thanh Đại, Trần Văn Sinh, "Rút gọn thuộc tính trong bảng quyết định đầy đủ theo tiếp cận xác suất phân lớp của hạt thông tin lân cận mờ", *Hội thảo quốc gia lần thứ XXV Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, Hà Nội, 2022.

[CT3] **Pham Minh Ngoc Ha**, Tran Thanh Dai, Nguyen Manh Hung, Hoang Tuan Dung, "A Novel Extension Method of VPFRS Mode for Attribute Reduction Problem in Numerial Decision Tables," *J. Comput. Sci. Cybern.*, no.1 Mar, 2024.

[CT4] **Pham Minh Ngoc Ha**, Nguyen Long Giang, Nguyen Manh Hung, Tran Thanh Dai, "Incremental Attribute Reduction in Rough Set for Fuzzy Decision Tables", *Vietnam Journal of Science and Technology*, 2024.

TÀI LIỆU THAM KHẢO

[1] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimedia Tools and Applications*, vol. 78, no. 3, 2019.

[2] B. Remeseiro and V. Bolon-Canedo, *A review of feature selection methods in medical applications*, 2019.

[3] Y. Jing, T. Li, H. Fujita, B. Wang, and N. Cheng, "An incremental attribute reduction method for dynamic data mining," *Information Sciences*, vol. 465, 2018.

[4] X. Rao, X. Yang, X. Yang, X. Chen, D. Liu, and Y. Qian, "Quickly calculating reduct: An attribute relationship based approach," *Knowledge-Based Systems*, vol. 200, p. 106 014, 2020.

[5] W. Wei and J. Liang, "Information fusion in rough set theory : An overview," *Information Fusion*, vol. 48, 2019.

[6] C. Zhang and J. Dai, "An incremental attribute reduction approach based on knowledge granularity for incomplete decision systems," *Granular Computing*, vol. 5, no. 4, 2020.

[7] P. Ray, S. S. Reddy, and T. Banerjee, "Various dimension reduction techniques for high dimensional data analysis: a review," *Artificial Intelligence Review*, vol. 54, no. 5, 2021.

[8] P. Dhal and C. Azad, "A comprehensive survey on feature selection in the various fields of machine learning," *Applied Intelligence*, vol. 52, no. 4, 2022.

[9] U. M. Khaire and R. Dhanalakshmi, *Stability of feature selection algorithm: A review*, 2022.

[10] C. Liu, B. Lin, J. Lai, and D. Miao, "An improved decision tree algorithm based on variable precision neighborhood similarity," *Information Sciences*, vol. 615, 2022.

[11] Q. Kong, X. Zhang, W. Xu, and B. Long, "A novel granular computing model based on three-way decision," *International Journal of Approximate Reasoning*, vol. 144, 2022.

[12] S. A. Shehab, A. Darwish, and A. E. Hassanien, "Water quality classification model with small features and class imbalance based on fuzzy rough sets," *Environment, Development and Sustainability*, 2023.

[13] T. Mahmood, J. Ahmmad, Z. Ali, and M. S. Yang, "Confidence Level Aggregation Operators Based on Intuitionistic Fuzzy Rough Sets With Application in Medical Diagnosis," *IEEE Access*, vol. 11, 2023.

[14] G. Zhang and W. Yao, "Fuzzy rough sets based on modular hemimetrics," *Soft Computing*, vol. 27, no. 8, 2023.

[15] B. Yu, Q. Zhu, Y. Fu, and M. Cai, "A twin logistic regression method based on attribute-oriented fuzzy rough set," *Journal of Intelligent and Fuzzy Systems*, vol. 44, no. 6, 2023.

[16] G. Lin, L. Xie, J. Li, J. Chen, and Y. Kou, "Local double quantitative fuzzy rough sets over two universes," *Applied Soft Computing*, vol. 145, 2023.

[17] J. Wang, X. Zhang, and C. Liu, "Grained matrix and complementary matrix: Novel methods for computing information descriptions in covering approximation spaces," *Information Sciences*, vol. 591, pp. 68–87, 2022.

[18] T. Zhou, H. L. Lu, H. L. Ren, and B. Q. Huo, *Survey on Attribute Reduction Algorithm of Rough Set*, 2021.

[19] C. Y. Wang and L. Wan, "New results on granular variable precision fuzzy rough sets based on fuzzy (co)implications," *Fuzzy Sets and Systems*, vol. 423, 2021.

[20] A. Theerens and C. Cornelis, "Fuzzy rough sets based on fuzzy quantification," *Fuzzy Sets and Systems*, vol. 473, 2023.

[21] X. Che, D. Chen, and J. Mi, "Learning Instance-Level Label Correlation Distribution for Multilabel Classification with Fuzzy Rough Sets," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 8, 2023.

[22] A. K. Tiwari, S. Shreevastava, T. Som, and K. K. Shukla, "Tolerance-based intuitionistic fuzzy-rough set approach for attribute reduction," *Expert Systems with Applications*, vol. 101, pp. 205–212, 2018.

[23] J. Chen, J. Mi, and Y. Lin, "A graph approach for fuzzy-rough feature selection," *Fuzzy Sets and Systems*, vol. 391, 2020.

[24] B. Yang, B. Q. Hu, and J. Qiao, "Three-way decisions with rough membership functions in covering approximation space," *Fundamenta Informaticae*, vol. 165, no. 2, 2019.

[25] X. Yang, H. Chen, H. Wang, *et al.*, "Feature Selection With Local Density-Based Fuzzy Rough Set Model for Noisy Data," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 5, 2023.

[26] X. Zhang, C. Mei, J. Li, Y. Yang, and T. Qian, "Instance and Feature Selection Using Fuzzy Rough Sets: A Bi-Selection Approach for Data Reduction," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 6, 2023.

[27] T. Yin, H. Chen, T. Li, Z. Yuan, and C. Luo, "Robust feature selection using label enhancement and β-precision fuzzy rough sets for multilabel fuzzy decision system," *Fuzzy Sets and Systems*, vol. 461, 2023.

[28] M. Hu, Y. Guo, D. Chen, E. C. Tsang, and Q. Zhang, "Attribute reduction based on neighborhood constrained fuzzy rough sets," *Knowledge-Based Systems*, vol. 274, 2023.

[29] X. Yang and Y. Yao, "Ensemble selector for attribute reduction," *Applied Soft Computing Journal*, vol. 70, 2018.

[30] L. Xie, G. Lin, J. Li, and Y. Lin, "A novel fuzzy-rough attribute reduction approach via local information entropy," *Fuzzy Sets and Systems*, vol. 473, 2023.

[31] W. Li, B. Yang, and J. Qiao, "(O, G)-granular variable precision fuzzy rough sets based on overlap and grouping functions," *Computational and Applied Mathematics*, vol. 42, no. 3, 2023.

[32] D. Huang, Y. Chen, F. Liu, and Z. Li, "Feature selection for multiset-valued data based on fuzzy conditional information entropy using iterative model and matrix operation," *Applied Soft Computing*, vol. 142, 2023.

[33] J. Liu, B. J. Zheng, L. M. Luo, J. S. Zhou, Y. Zhang, and Z. T. Yu, "Ontology representation and mapping of common fuzzy knowledge," *Neurocomputing*, vol. 215, 2016.

[34] R. Joshi, "Multi-criteria decision making based on novel fuzzy knowledge measures," *Granular Computing*, vol. 8, no. 2, 2023.

[35] P. Sun and L. Gu, "Fuzzy knowledge graph system for artificial intelligence-based smart education," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, 2021.

[36] J. Qiao and B. Q. Hu, "Granular variable precision L-fuzzy rough sets based on residuated lattices," *Fuzzy Sets and Systems*, vol. 336, 2018.

[37] M. Aggarwal, "Probabilistic variable precision fuzzy rough sets," *IEEE Transactions on Fuzzy Systems*, vol. 24, no. 1, 2016.

[38] C. Y. Wang and B. Q. Hu, "Granular variable precision fuzzy rough sets with general fuzzy relations," *Fuzzy Sets and Systems*, vol. 275, 2015.

[39] Z. Shi, S. Xie, and L. Li, "Generalized fuzzy neighborhood system-based multigranulation variable precision fuzzy rough sets with double TOPSIS method to MADM," *Information Sciences*, vol. 643, 2023.

[40] H. Jiang, J. Zhan, and D. Chen, "PROMETHEE II method based on variable precision fuzzy rough sets with fuzzy neighborhoods," *Artificial Intelligence Review*, vol. 54, no. 2, 2021.

[41] Z. ao Xue, M. meng Jing, Y. xiang Li, *et al.*, "Variable precision multi-granulation covering rough intuitionistic fuzzy sets," *Granular Computing*, 2022.

[42] B. W. Fang and B. Q. Hu, "Granular fuzzy rough sets based on fuzzy implicators and coimplicators," *Fuzzy Sets and Systems*, vol. 359, pp. 112–139, 2019.

[43] J. Zhan, H. Jiang, and Y. Yao, "Covering-based variable precision fuzzy rough sets with PROMETHEE-EDAS methods," *Information Sciences*, vol. 538, no. 2, pp. 1093–1126, 2020.

[44] T. Feng and J. S. Mi, "Variable precision multigranulation decision-theoretic fuzzy rough sets," *Knowledge-Based Systems*, vol. 91, 2016.

[45] D. Zou, Y. Xu, L. Li, and Z. Ma, "Novel variable precision fuzzy rough sets and three-way decision model with three strategies," *Information Sciences*, vol. 629, 2023.

[46] Y. Liu, Y. Lin, and H. H. Zhao, "Variable precision intuitionistic fuzzy rough set model and applications based on conflict distance," *Expert Systems*, vol. 32, no. 2, 2015.

[47] C. Wang, M. Shao, Q. He, Y. Qian, and Y. Qi, "Feature subset selection based on fuzzy neighborhood rough sets," *Knowledge-Based Systems*, vol. 111, pp. 173–179, 2016.

[48] S. Zhao, E. C. Tsang, and D. Chen, "The Model of Fuzzy Variable Precision Rough Sets," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 2, 2009.

[49] H. Y. Zhang, H. J. Song, and S. Y. Yang, "Feature selection based on generalized variable-precision (σ) -fuzzy granular rough set model over two universes," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 5, 2019.

[50] C. Cornelis and R. Jensen, "A Noise-tolerant approach to fuzzy-rough feature selection," in *IEEE International Conference on Fuzzy Systems*, 2008.

[51] X. Xin, J. Song, and W. Peng, "Intuitionistic fuzzy three-way decision model based on the three-way granular computing method," *Symmetry*, vol. 12, no. 7, p. 1068, 2020.

[52] S. P. Tiwari, A. P. Singh, and S. Pandey, "Note on 'Intuitionistic L-fuzzy Rough Sets, Intuitionistic L-fuzzy Preorders and Intuitionistic L-fuzzy Topologies'," *Fuzzy Information and Engineering*, vol. 13, no. 2, pp. 196–198, 2021.

[53] H. Garg and G. Kaur, "Novel distance measures for cubic intuitionistic fuzzy sets and their applications to pattern recognitions and medical diagnosis," *Granular Computing*, vol. 5, no. 2, pp. 169–184, 2020.

[54] T. T. Nguyen, N. L. Giang, D. T. Tran, *et al.*, "A novel filter-wrapper algorithm on intuitionistic fuzzy set for attribute reduction from decision tables," *International Journal of Data Warehousing and Mining*, vol. 17, no. 4, pp. 67–100, 2021.

[55] L. Dong, R. Wang, and D. Chen, "Incremental feature selection with fuzzy rough sets for dynamic data sets," *Fuzzy Sets and Systems*, vol. 467, 2023.

[56] W. Ding, T. Qin, X. Shen, *et al.*, "Parallel incremental efficient attribute reduction algorithm based on attribute tree," *Information Sciences*, vol. 610, 2022.

[57] A. Kumar and P. S. Sai Prasad, "Incremental fuzzy rough sets based feature subset selection using fuzzy min-max neural network preprocessing," *International Journal of Approximate Reasoning*, vol. 139, pp. 69–87, 2021.

[58] C. Luo, T. Li, H. Chen, H. Fujita, and Z. Yi, "Incremental rough set approach for hierarchical multicriteria classification," *Information Sciences*, vol. 429, 2018.

[59] W. Shu and H. Shen, "Incremental feature selection based on rough set in dynamic incomplete data," *Pattern Recognition*, vol. 47, no. 12, 2014.

[60] X. Zhang, C. Mei, D. Chen, Y. Yang, and J. Li, "Active Incremental Feature Selection Using a Fuzzy-Rough-Set-Based Information Entropy," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 901–915, 2020.

[61] W. Wei, X. Wu, J. Liang, J. Cui, and Y. Sun, "Discernibility matrix based incremental attribute reduction for dynamic data," *Knowledge-Based Systems*, vol. 140, 2018.

[62] P. Bermejo, J. A. Gámez, and J. M. Puerta, "Speeding up incremental wrapper feature subset selection with Naive Bayes classifier," *Knowledge-Based Systems*, vol. 55, 2014.

[63] G. Hao, L. Longshu, Y. Chuanjian, and D. Jian, "Incremental reduction algorithm with acceleration strategy based on conflict region," *Artificial Intelligence Review*, vol. 51, no. 4, 2019.

[64] C. Hu, L. Zhang, B. Wang, Z. Zhang, and F. Li, "Incremental updating knowledge in neighborhood multigranulation rough sets under dynamic granular structures," *Knowledge-Based Systems*, vol. 163, 2019.

[65] L. Z. Yin, C. H. Yang, X. L. Wang, and W. H. Gui, "An incremental algorithm for attribute reduction based on labeled discernibility matrix," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 40, no. 3, pp. 397–404, 2014.

[66] B. Sang, H. Chen, L. Yang, T. Li, and W. Xu, "Incremental Feature Selection Using a Conditional Entropy Based on Fuzzy Dominance Neighborhood Rough Sets," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 6, pp. 1683–1697, 2022.

[67] J. Hu, T. Li, C. Luo, H. Fujita, and S. Li, "Incremental fuzzy probabilistic rough sets over two universes," *International Journal of Approximate Reasoning*, vol. 81, 2017.

[68] F. Wang, J. Liang, and Y. Qian, "Attribute reduction: A dimension incremental strategy," *Knowledge-Based Systems*, vol. 39, pp. 95–108, 2013.

[69] X. Xie and X. Qin, "A novel incremental attribute reduction approach for dynamic incomplete decision systems," *International Journal of Approximate Reasoning*, vol. 93, 2018.

[70] X. Yang, M. Li, H. Fujita, D. Liu, and T. Li, "Incremental rough reduction with stable attribute group," *Information Sciences*, vol. 589, 2022.

[71] S. Li and T. Li, "Incremental update of approximations in dominance-based rough sets approach under the variation of attribute values," *Information Sciences*, vol. 294, 2015.

[72] W. Shu, W. Qian, and Y. Xie, "Incremental feature selection for dynamic hybrid data using neighborhood rough set," *Knowledge-Based Systems*, vol. 194, no. 105516, p. 105 516, 2020.

[73] A. Zeng, T. Li, D. Liu, J. Zhang, and H. Chen, "A fuzzy rough set approach for incremental feature selection on hybrid information systems," *Fuzzy Sets and Systems*, vol. 258, 2015.

[74] N. Nandhini and K. Thangadurai, "An incremental rough set approach for faster attribute reduction," *International Journal of Information Technology (Singapore)*, vol. 15, no. 2, 2023.

[75] C. C. Huang, T. L. Tseng, Y. N. Fan, and C. H. Hsu, "Alternative rule induction methods based on incremental object using rough set theory," *Applied Soft Computing Journal*, vol. 13, no. 1, 2013.

[76] N. L. Giang, L. H. Son, T. T. Ngan, *et al.*, "Novel Incremental Algorithms for Attribute Reduction from Dynamic Decision Tables Using Hybrid Filter-Wrapper with Fuzzy Partition Distance," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 858–873, 2020.

[77] Y. Jing, T. Li, H. Fujita, Z. Yu, and B. Wang, "An incremental attribute reduction approach based on knowledge granularity with a multi-granulation view," *Information Sciences*, vol. 411, pp. 23–38, 2017.

[78] Y. Yang, D. Chen, X. Zhang, Z. Ji, and Y. Zhang, "Incremental feature selection by sample selection and feature-based accelerator[Formula presented]," *Applied Soft Computing*, vol. 121, 2022.

[79] D. Xia, G. Wang, Q. Zhang, J. Yang, S. Li, and M. Gao, "Incremental Approximation Feature Selection With Accelerator for Rough Fuzzy Sets by Knowledge Distance," *IEEE Transactions on Fuzzy Systems*, vol. 31, no. 11, 2023.

[80] F. Wang, W. Wei, and J. Liang, "A group incremental approach for feature selection on hybrid data," *Soft Computing*, vol. 26, no. 8, pp. 3663–3677, 2022.

[81] S. Li, T. Li, and D. Liu, "Incremental updating approximations in dominance-based rough sets approach under the variation of the attribute set," *Knowledge-Based Systems*, vol. 40, 2013.

[82] D. Liu, T. Li, and J. Zhang, "Incremental updating approximations in probabilistic rough sets under the variation of attributes," *Knowledge-Based Systems*, vol. 73, 2015.

[83] M. S. Raza and U. Qamar, "An incremental dependency calculation technique for feature selection using rough sets," *Information Sciences*, vol. 343-344, 2016.

[84] Y. Yang, D. Chen, H. Wang, and X. Wang, "Incremental Perspective for Feature Selection Based on Fuzzy Rough Sets," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 3, 2018.

[85] D. Liu, T. Li, and J. Zhang, "A rough set-based incremental approach for learning knowledge in dynamic incomplete information systems," *International Journal of Approximate Reasoning*, vol. 55, no. 8, 2014.

[86] Y. Yang, D. Chen, H. Wang, E. C. Tsang, and D. Zhang, "Fuzzy rough set based incremental attribute reduction from dynamic data with sample arriving," *Fuzzy Sets and Systems*, vol. 312, 2017.

[87] Y. Jing, T. Li, C. Luo, S. J. Horng, G. Wang, and Z. Yu, "An incremental approach for attribute reduction based on knowledge granularity," *Knowledge-Based Systems*, vol. 104, 2016.

[88] H. Chen, T. Li, D. Ruan, J. Lin, and C. Hu, "A rough-set-based incremental approach for updating approximations under dynamic maintenance environments," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 2, 2013.

[89] J. Liang, F. Wang, C. Dang, and Y. Qian, "A group incremental approach to feature selection applying rough set technique," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 2, 2014.

[90] Y. Jing, T. Li, J. Huang, and Y. Zhang, "An incremental attribute reduction approach based on knowledge granularity under the attribute generalization," *International Journal of Approximate Reasoning*, vol. 76, 2016.

[91] Y. Liu, L. Zheng, Y. Xiu, *et al.*, "Discernibility matrix based incremental feature selection on fused decision tables," *International Journal of Approximate Reasoning*, vol. 118, pp. 1–26, Mar. 2020.

[92] F. Pourpanah, Y. Shi, C. P. Lim, Q. Hao, and C. J. Tan, "Feature selection based on brain storm optimization for data classification," *Applied Soft Computing Journal*, vol. 80, pp. 761–775, 2019.

[93] UCI, *Uc irvine machine learning repository*, Accessed on 26 March 2023, 2023.

[94] F. Asdaghi and A. Soleimani, "An effective feature selection method for web spam detection," *Knowledge-Based Systems*, vol. 166, 2019.

[95] M. R. Keyvanpour, M. Barani Shirzad, and F. Heydarian, "Android malware detection applying feature selection techniques and machine learning," *Multimedia Tools and Applications*, vol. 82, no. 6, 2023.

[96] N. Dabas, P. Ahlawat, and P. Sharma, "An Effective Malware Detection Method Using Hybrid Feature Selection and Machine Learning Algorithms," *Arabian Journal for Science and Engineering*, vol. 48, no. 8, 2023.

[97] A. K. Sangaiah, A. Javadpour, F. Ja'fari, P. Pinto, W. Zhang, and S. Balasubramanian, "A hybrid heuristics artificial intelligence feature selection for intrusion detection classifiers in cloud of things," *Cluster Computing*, vol. 26, no. 1, 2023.

[98] B. Yu, Y. Hu, Y. Kang, and M. Cai, "A novel variable precision rough set attribute reduction algorithm based on local attribute significance," *International Journal of Approximate Reasoning*, vol. 157, 2023.

[99] B. Sang, H. Chen, L. Yang, J. Wan, T. Li, and W. Xu, "Feature Selection Considering Multiple Correlations based on Soft Fuzzy Dominance Rough Sets for Monotonic Classification," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 12, pp. 5181–5195, 2022.