DOI: 10.64302/joshusc.v33n1.1353

THUẬT TOÁN RÚT GỌN THUỘC TÍNH DỰA VÀO CẮT TỈA MỀM VÀ TẬP THÔ LÂN CẬN TRỌNG SỐ TỔNG QUÁT

Trần Duy Anh, Lê Mạnh Thạnh, Đoàn Thị Hồng Phước, Nguyễn Ngọc Thủy*

Khoa Công nghệ thông tin, Trường Đại học Khoa học, Đại học Huế

*Email: nnthuy.cs@hueuni.edu.vn

Ngày nhận bài: 20/10/2025; ngày hoàn thành phản biện: 25/11/2025; ngày duyệt đăng: 6/12/2025

TÓM TẮT

Rút gọn thuộc tính là một nhiệm vụ trọng tâm trong lý thuyết tập thô, nhằm loại bỏ các thuộc tính dư thừa mà vẫn duy trì khả năng phân lớp của hệ quyết định. Trong khuôn khổ tập thô lân cận trọng số tổng quát (GWNRS), thuật toán heuristic hiện có đã cho thấy hiệu quả nhất định nhưng vẫn chịu chi phí tính toán cao khi áp dụng cho dữ liệu quy mô lớn. Để khắc phục hạn chế này, chúng tôi đề xuất thuật toán SP-GWNRS mới dựa trên cơ chế cắt tỉa mềm. Mỗi thuộc tính được gắn với một bộ đếm theo dõi số lần đóng góp tiêu cực vào độ phụ thuộc, qua đó thuật toán có thể thích ứng giữa tìm kiếm và cắt tỉa nhằm giảm các đánh giá dư thừa. Thêm nữa, một giai đoạn wrapper cũng được bổ sung để chọn tập rút gọn cuối cùng tốt nhất dựa trên độ chính xác phân lớp. Kết quả thực nghiệm trên các bộ dữ liệu chuẩn cho thấy rằng SP-GWNRS trả về tập rút gọn có kích thước tương đương với thuật toán gốc, nhưng vượt trội về độ chính xác.

Từ khóa: Lý thuyết tập thô; Tập thô lân cận trọng số tổng quát; Rút gọn thuộc tính; Cắt tỉa mềm; Hệ thông tin quyết định.

1. GIỚI THIỆU

Lý thuyết tập thô được giới thiệu bởi Pawlak trong [1] là một công cụ toán học hiệu quả cho việc khám phá tri thức, chọn lọc thuộc tính và phân tích dữ liệu. Một trong những nhiệm vụ trung tâm của lý thuyết tập thô là rút gọn thuộc tính nhằm loại bỏ các thuộc tính dư thừa, trong khi vẫn duy trì được khả năng phân lớp của hệ quyết định. Trong nhiều thập kỷ qua, đã có rất nhiều mở rộng được đề xuất để tăng cường khả năng ứng dụng của tập thô đối với dữ liệu thực tế, chẳng hạn như tập thô lân cận, tập thô lân cận mềm, và tập thô mờ [2–4].

Trong số các mở rộng này, tập thô lân cận và các biến thể có trọng số của nó đã thu hút nhiều sự quan tâm, vì chúng cho phép xử lý hiệu quả các trường hợp dữ liệu số liên tục với số chiều lớn thông qua việc xây dựng các hạt thông tin lân cận ([5–8]). Gần đây, chúng tôi đã đề xuất tập thô lân cận có trọng số tổng quát (GWNRS) như một công cụ mạnh mẽ hơn. GWNRS kết hợp trọng số trên từng thuộc tính và trọng số cho từng đối tượng trong lân cận. Qua đó cải thiện tính linh hoạt và độ chính xác trong bài toán rút gọn thuộc tính theo chiến lược tham lam [9]. Mặc dù đạt hiệu quả phân lớp cao từ tập con thuộc tính được lựa chọn, nhưng GWNRS vẫn gặp phải vấn đề về chi phí tính toán cao, đặc biệt khi áp dụng cho các tập dữ liệu có số lượng lớn đối tượng và thuộc tính. Trong thực tế, chiến lược lựa chọn tham lam thường dẫn đến việc đánh giá lặp lại các thuộc tính ít đóng góp, thậm chí có tác động tiêu cực đến mức độ phụ thuộc, gây ra tính toán dư thừa và làm giảm khả năng mở rộng. Do đó, việc thiết kế những thuật toán hiệu quả hơn là một nhu cầu cấp thiết, nhằm xử lý các tập dữ liệu lớn mà không làm suy giảm độ chính xác.

Cụ thể, chúng tôi đề xuất một thuật toán heuristic mới cho bài toán rút gọn thuộc tính dựa trên cơ chế cắt tỉa mềm (soft pruning) và tập thô lân cận có trọng số tổng quát, gọi là SP-GWNRS. Thuật toán này được thiết kế gồm hai giai đoạn:

- Ở giai đoạn thứ nhất, cơ chế cắt tỉa mềm được sử dụng để thích ứng trong quá trình lựa chọn, giúp loại bỏ tạm thời các thuộc tính mang ít thông tin, đồng thời cho phép khôi phục lại chúng sau một số vòng lặp. Chiến lược này giúp cân bằng giữa việc cắt tỉa và khám phá, giảm đáng kể các phép đánh giá dư thừa trong khi vẫn duy trì được quá trình xây dựng tập rút gọn ổn định.
- Ở giai đoạn thứ hai, một pha đánh giá wrapper được đưa vào nhằm chọn ra tập rút gọn cuối cùng từ các ứng viên trung gian bằng cách tối đa hóa độ chính xác phân lớp. Từ đó đảm bảo cả hiệu quả tính toán và hiệu năng dự đoán của rút gọn thu được.

Những đóng góp chính của bài báo này được tóm tắt như sau:

(1) Đề xuất thuật toán SP-GWNRS mới kết hợp chiến lược cắt tỉa mềm với cơ chế kết hợp hai pha filter và wrapper, giúp cải thiện đồng thời hiệu quả tính toán và hiệu năng phân lớp.

(2) Thực hiện các thí nghiệm mở rộng trên các tập dữ liệu chuẩn, cho thấy SP-GWNRS tìm ra các tập rút gọn có chất lượng và nâng cao độ chính xác phân lớp so với thuật toán GWNRS gốc.

Phần còn lại của bài báo được tổ chức như sau: Phần 2 trình bày các khái niệm cơ bản của tập thô lân cận có trọng số tổng quát. Phần 3 giới thiệu thuật toán SP-GWNRS được đề xuất. Phần 4 trình bày kết quả thực nghiệm cùng một số so sánh, đánh giá. Cuối cùng, Phần 5 đưa ra kết luận và định hướng nghiên cứu tương lai.

2. TẬP THÔ LÂN CẬN TRỌNG SỐ TỔNG QUÁT

Trong phần này, chúng tôi trình bày các khái niệm cơ bản như hệ thông tin, bảng quyết định, sau đó nhắc lại định nghĩa và một số tính chất của các tập thô lân cận trọng số tổng quát [9].

Một hệ thông tin được định nghĩa là một cặp $I = (U, A)$, trong đó U là tập hữu hạn khác rỗng các đối tượng và A là tập hữu hạn khác rỗng các thuộc tính. Mỗi thuộc tính $a \in A$ xác định một ánh xạ $a: U \to V_a$, với V_a là tập giá trị của thuộc tính a. Với mỗi thuộc tính $a \in A$ và đối tượng $u \in U$, giá trị của đối tượng u trên thuộc tính a được ký hiệu là $a(u)$.

Trong trường hợp đặc biệt khi A được chia thành hai tập con rời nhau, C và D, trong đó C được gọi là tập thuộc tính điều kiện và D được gọi là tập thuộc tính quyết định, thì hệ thông tin I sẽ được gọi là *hệ thông tin quyết định* (hoặc ngắn gọn hơn gọi là *bảng quyết định*) và được ký hiệu bởi $S = (U, C \cup D)$.

Cho tập con thuộc tính $B \subseteq C$ và u, v là hai đối tượng, khoảng cách giữa u và v trên tập thuộc tính B, ký hiệu $\Delta_B(u, v)$, được xác định bởi

$$\Delta_B(u, v) = \left(\sum_{a \in B} |a(u) - a(v)|^p \right)^{\frac{1}{p}}, \quad (1)$$

trong đó $\Delta_B(u, v)$ được gọi là khoảng cách Manhattan nếu $p = 1$, khoảng cách Euclidean nếu $p = 2$, và khoảng cách Chebychev nếu $p = \infty$.

Xét hệ thông tin quyết định $S = (U, C \cup D)$, một tập con thuộc tính $B \subseteq C$ và một số nguyên dương k, mỗi đối tượng $u \in U$ sẽ xác định *một hạt thông tin k-lân cận gần nhất*, ký hiệu $N_B^k(u)$, chứa k đối tượng có khoảng cách gần nhất đến u.

Cho tập con thuộc tính $B \subseteq C$ và hai đối tượng u, v, khoảng cách có trọng số giữa u và v trên tập thuộc tính B, ký hiệu $\Delta_B^{w^{att}}(u, v)$, được xác định bởi

$$\Delta_B^{w^{att}}(u, v) = \left(\sum_{a \in B} w^{att}(a) \cdot |a(u) - a(v)|^p \right)^{\frac{1}{p}}, \quad (2)$$

trong đó $w^{att}(a)$ là trọng số của thuộc tính $a \in B$.

Khi đó, hạt thông tin ε-lân cận có trọng số thuộc tính của đối tượng u được định nghĩa như sau:

$$GWN_B^{\varepsilon}(u) = \left\{ v \in U : \Delta_B^{w^{att}}(u, v) \leq \varepsilon \right\}, \quad (3)$$

với ε > 0 là bán kính lân cận.

Tiếp theo, cho $u \in U$, hạt thông tin lân cận trọng số thuộc tính của đối tượng u được xác định bởi:

$$\Theta_B(u) = N_B^k(u) \cap GWN_B^\varepsilon(u). \quad (4)$$

Có thể thấy rằng $\Theta_B(u)$ là tập chứa nhiều nhất k đối tượng gần nhất trong bán kính ε. Hiển nhiên, $\Theta_B(u)$ có thể là tập rỗng nếu u là đối tượng nhiễu. Do đó, họ tất cả các hạt thông tin lân cận trọng số thuộc tính, $F(B) = \{\Theta_B(u): u \in U\}$, có thể không hình thành lên một phủ trên U.

Rõ ràng, mỗi đối tượng trong $\Theta_B(u)$ đóng một vai trò khác nhau trong việc đánh giá, quyết định đối với u. Do đó, chúng ta cần gán trọng số cho các đối tượng trong $\Theta_B(u)$. Giả sử rằng $\Theta_B(u) = \{u_1, u_2, \dots, u_{|\Theta_B(u)|}\}$, khi đó chúng ta gán các trọng số tương ứng là $w^o(u_1)$, $w^o(u_2)$, ..., $w^o(u_{|\Theta_B(u)|})$) với $w^o(u_i) \in [0,1]$.

Như vậy, một hạt thông tin lân cận trọng số tổng quát được định nghĩa là:

$$\widetilde{\Theta_B}(u) = \left\{\frac{u_1}{w^o(u_1)}, \frac{u_2}{w^o(u_2)}, \dots, \frac{u_{|\Theta_B(u)|}}{w^o\left(u_{|\Theta_B(u)|}\right)}\right\}. \quad (5)$$

Dễ thấy $\widetilde{\Theta_B}(u)$ có thể được xem như một tập mờ, trong đó $w^o(u_i)$ chính là độ thành viên của u_i thuộc vào tập $\Theta_B(u)$. Chúng ta ký hiệu $\tilde{F}(B) = \{\widetilde{\Theta_B}(u): u \in U\}$ là họ tất cả các hạt thông tin lân cận trọng số tổng quát trên U.

Bây giờ, chúng ta sẽ trình bày các khái niệm cơ sở của tập thô GWNRSs. Cho hệ thông tin quyết định $S = (U, C \cup D)$, một tập con thuộc tính $B \subseteq C$, và một tập đối tượng $X \subseteq U$, Xấp xỉ dưới và trên của X theo B được định nghĩa là:

$$\underline{GW_B}(X) = \left\{u \in U : \frac{|\widetilde{\Theta_B}(u) \cap X|}{|\widetilde{\Theta_B}(u)|} \geq \alpha\right\}, \quad (6)$$

và

$$\overline{GW_B}(X) = \left\{u \in U : \frac{|\widetilde{\Theta_B}(u) \cap X|}{|\widetilde{\Theta_B}(u)|} > \beta\right\}, \quad (7)$$

trong đó $0 \leq \beta < \alpha \leq 1$.

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

$$\gamma_B^{GW}(D) = \frac{|GWPOS_B(D)|}{|U|}, \quad (9)$$

trong đó $GWPOS_B(D)$ là miền khẳng định của D theo B, được xác định bởi:

$$GWPOS_B(D) = \bigcup_{X \in U/D} \underline{GW_B}(X). \quad (10)$$

3. RÚT GỌN THUỘC TÍNH VỚI GWNRS SỬ DỤNG CẮT TỈA MỀM

Phần này trình bày định nghĩa rút gọn theo tiếp cận tập thô lân cận trọng số tổng quát GWNRS. Sau đó, chúng tôi chỉ ra nhược điểm cần khắc phục của trong thuật toán rút gọn thuộc tính gốc trong [9] và đề xuất ý tưởng cắt tỉa mềm kết hợp với cách tiếp cận filter-wrapper để phát triển thuật toán mới SP-GWNRS.

Cho $S = (U, C \cup D)$ là bảng quyết dịnh. Một tập con thuộc tính $R \subseteq C$ được gọi là *một rút gọn của* C trong việc quyết định D nếu nó thỏa mãn các tính chất sau:

i) $\gamma_R^{GW}(D) \geq \gamma_C^{GW}(D)$,
ii) $\forall a \in R, \gamma_{R\setminus\{a\}}^{GW}(D) < \gamma_R^{GW}(D)$.

Trong các thuật toán hueristic tìm kiếm nhanh các rút gọn, việc xác định mức độ quan trọng của một thuộc tính đóng vai trò then chốt. Một cách hình thức, điều này có thể được định nghĩa như sau:

Cho tập con thuộc tính $B \subseteq C$ và thuộc tính $a \in C\setminus B$, *độ quan trọng của thuộc tính* a đối với B được định nghĩa là:

$$SIG^{GW}(a, B, D) = \gamma_{B\cup\{a\}}^{GW}(D) - \gamma_B^{GW}(D). \quad (11)$$

Có thể thấy rằng $SIG^{GW}(a, B, D)$ đo mức độ gia tăng độ phụ thuộc khi thuộc tính a được thêm vào B. Giá trị độ đo này càng lớn, phản ánh đóng góp của thuộc tính a vào việc quyết định các đối tượng trong vũ trụ U càng lớn.

Để tìm tập rút gọn, trong [9], các tác giả đã đề xuất thuật toán heuristic rút gọn thuộc tính dựa vào GWNRS. Mặc dù thuật toán này hoạt động hiệu quả đối với các tập dữ liệu có quy mô nhỏ và trung bình, song khi áp dụng cho dữ liệu có số chiều lớn, nó vẫn tồn tại hạn chế về chi phí tính toán. Lý do là bởi một số thuộc tính có thể bị đánh giá lặp lại nhiều lần mặc dù không cải thiện được độ phụ thuộc, dẫn đến tính toán dư thừa. Những thuộc tính như vậy có thể chi phối không gian tìm kiếm, làm chậm quá trình hội tụ. Do đó, cần thiết phải xây dựng một chiến lược cắt tỉa có khả năng thích ứng động, nhằm loại bỏ tạm thời các thuộc tính không hiệu quả trong quá trình tìm kiếm.

Cụ thể, để khắc phục các vấn đề nêu trên, chúng tôi đề xuất một thuật toán rút gọn thuộc tính heuristic cải tiến dựa trên cơ chế cắt tỉa mềm trong khuôn khổ GWNRS, gọi là SP-GWNRS (Soft Pruning – Generalized Weighted Neighborhood Rough Set).

Ý tưởng cốt lõi của thuật toán là gắn với mỗi thuộc tính một bộ đếm (fail_counter), phản ánh số lần thuộc tính đó không đóng góp tích cực vào tập rút gọn. Một thuộc tính a được coi là "không đóng góp tích cực" nếu việc thêm thuộc tính này vào tập rút gọn RED không làm tăng độ phụ thuộc, nghĩa là $SIG^{GW}(a, RED, D) < 0$. Khi bộ đếm của một thuộc tính đạt đến ngưỡng *T*, thuộc tính này sẽ được tạm thời loại bỏ khỏi quá trình xem xét. Sau mỗi *N* vòng lặp, các bộ đếm được giảm đi một lượng τ, cho phép các thuộc tính bị loại trước đó có thể được xem xét lại nếu giá trị bộ đếm của chúng

giảm xuống dưới ngưỡng T. Cơ chế cắt tỉa mềm động này giúp nâng cao hiệu quả xử lý đối với các tập dữ liệu có quy mô lớn và nhiều chiều, đồng thời duy trì khả năng khám phá các thuộc tính tiềm năng trong quá trình rút gọn. Thuật toán cụ thể được trình bày như dưới đây.

Algorithm 1. Thuật toán filter-wrapper cho rút gọn thuộc tính dựa vào cắt tỉa mềm và tập thô lân cận trọng số tổng quát(SP-GWNRS)

Đầu vào: Hệ thông tin quyết định $S = (U, C \cup D)$, các tham số xây dựng hạt thông tin ε, k, α, β và các tham số cắt tỉa T, N, τ.

Đầu ra: Tập thuộc tính rút gọn RED

// Pha filter: Rút gọn heuristic với cắt tỉa mềm

1. Khởi tạo RED $\leftarrow \emptyset$, $\gamma_{RED}^{GW}(D) \leftarrow 0$, $iteration \leftarrow 0$;
2. **For every** $a \in C$:
3. Khởi tạo $fail\_counter[a] \leftarrow 0$ và $soft\_prune[a] \leftarrow False$;
4. **While** $C \backslash RED \neq \emptyset$:
5. **For every** $a \in C \backslash RED$:
6. **If** $soft\_prune[a] = True$:
7. $SIG^{GW}(a, RED, D) \leftarrow -1$;
8. **continue;**
9. $SIG^{GW}(a, RED, D) \leftarrow \gamma_{RED \cup \{a\}}^{GW}(D) - \gamma_{RED}^{GW}(D)$;
10. **If** $SIG^{GW}(a, RED, D) < 0$:
11. $fail\_counter[a] \mathrel{+}= (1 - SIG^{GW}(a, RED, D))$;
12. **If** $fail\_counter[a] \geq T$: $soft\_prune[a] \leftarrow True$;
13. **Else**:
14. $fail\_counter[a] \leftarrow 0$;
15. Tìm a_0 sao cho $SIG^{GW}(a_0, RED, D) = \max_{a \in C \backslash RED} SIG^{GW}(a, RED, D)$
16. **If** $SIG^{GW}(a_0, RED, D) \geq 0$:
17. $RED \leftarrow RED \cup \{a_0\}$;
18. Lưu ứng viên rút gọn $S_{|RED|} = RED$;
19. **Else:**
20. **break;**
21. $iteration \leftarrow iteration + 1$;
22. **If** $iteration \bmod N = 0$:
23. **For every** $a \in C \setminus RED$:
24. $fail\_counter[a] = max(0, fail\_counter[a] - \tau)$;
25. **If** $fail\_counter[a] <$ T: $soft\_prune[a] \leftarrow False$;

// Pha Wrapper

26: **For each** ứng viên S_i:

27: Đánh giá độ chính xác phân lớp của S_i với một bộ phân lớp;

28: Chọn S_{best} là tập thuộc tính có độ chính xác phân lớp cao nhất.

29: **Return** $RED = S_{best}$

Dễ thấy, chi phí tính toán của thuật toán SP-GWNRS chủ yếu bị chi phối bởi bước đánh giá độ quan trọng của thuộc tính. Bước này yêu cầu xây dựng các hạt thông tin lân cận trọng số tổng quát, với thời gian $O(|U|^2)$ cho mỗi lần đánh giá. Do đó, độ phức tạp thời gian của SP-GWNRS vẫn tương đương với thuật toán gốc. Tuy nhiên, việc tích hợp cơ chế cắt tỉa mềm đã giảm đáng kể các phép đánh giá dư thừa, qua đó cải thiện hiệu suất thực tế của quá trình tính toán. Cụ thể, trong thuật toán GWNRS gốc, tất cả $|C|$ thuộc tính có thể được xem xét tại mỗi vòng lặp, và quá trình này được lặp lại tối đa |RED| lần (với $|RED| \leq |C|$), dẫn đến độ phức tạp tổng thể là $O(|C|^2 \cdot |U|^2)$. Ngược lại, trong thuật toán SP-GWNRS, cơ chế cắt tỉa mềm cho phép loại bỏ tạm thời các thuộc tính không mang thông tin, do đó trung bình chỉ có $\varrho \cdot |C|$ thuộc tính được đánh giá tại mỗi vòng lặp, trong đó $0 < \varrho \leq 1$. Vì vậy, độ phức tạp thời gian của SP-GWNRS có thể được biểu diễn như sau: $O(\varrho \cdot |C|^2 \cdot |U|^2)$. Do ϱ thường nhỏ hơn đáng kể so với 1 trong các tập dữ liệu có số chiều cao, nên SP-GWNRS giảm đáng kể chi phí tính toán.

4. THỰC NGHIỆM VÀ PHÂN TÍCH, ĐÁNH GIÁ

Để đánh giá hiệu quả của thuật toán được đề xuất, chúng tôi đã tiến hành một loạt thí nghiệm trên các tập dữ liệu khác nhau. Mục tiêu của các thí nghiệm này là so sánh hiệu năng của SP-GWNRS với phương pháp rút gọn thuộc tính heuristic gốc dựa trên GWNRS. Cả hai thuật toán đều được thực thi trong cùng điều kiện tham số, bao gồm kích thước lân cận (k) và bán kính (ε). Đối với SP-GWNRS, các tham số bổ sung (T, N, τ) được điều chỉnh thực nghiệm trong một phạm vi hợp lý nhằm đạt được hiệu suất tối ưu. Cụ thể trong các thí nghiệm, tham số k được điều chỉnh theo số lượng đối tượng ($|U|$) như sau: Khi ($100 \leq |U| \leq 1000$), k được thay đổi từ ($0.01|U|$) đến ($0.15|U|$), với bước tăng $0.01|U|$. Khi $|U| < 100$, k được chọn trong khoảng từ 1 đến 15, với bước tăng là 1. Khi $|U| > 1000$, k được điều chỉnh trong khoảng $0.001|U|$ đến $0.015|U|$, với bước tăng $0.001|U|$. Tham số ε được thay đổi trong khoảng từ 0.1 đến 1.0, với bước tăng 0.1. Các tham số (T, N, τ) được thiết lập cụ thể là $T = 2$, $N = max(2, 2*log(|C|))$ và $\tau = 1$. Ngoài ra, để đánh giá độ chính xác phân lớp của tập rút gọn thu được, chúng tôi sử dụng hai bộ phân lớp Naive Bayes (NB) và KNN (với $K = 3$). Những kết quả thực nghiệm này được dùng để phân tích và so sánh hiệu quả của SP-GWNRS với thuật toán gốc trên các khía cạnh về độ chính xác phân lớp và kích thước rút gọn.

Tất cả hai thuật toán được triển khai trên Python 3.10.11 và được thực thi trên máy tính PC với Intel(R) Core(TM) i7-12700 2.10 GHz, RAM 16.0 GB.

Các tập dữ liệu được sử dụng trong thí nghiệm là các hệ quyết định chuẩn có kích thước và số chiều khác nhau. Bảng 1 trình bày các đặc trưng chính của chúng, bao

gồm: số lượng đối tượng $|U|$, số lượng thuộc tính điều kiện $|C|$, số lượng lớp quyết định, và loại dữ liệu tương ứng. Các tập dữ liệu này được tải về từ các kho dữ liệu về khai phá dữ liệu và học máy phổ biến bao gồm: Kho dữ liệu về học máy UCI https://archive.ics.uci.edu/datasets, Kho dữ liệu mở về học máy OpenML: https://openml.org/search?type=data%26sort=runs%26status=active, và Kho dữ liệu mở Kaggle: https://www.kaggle.com/datasets/.

Hiệu quả so sánh giữa SP-GWNRS và GWNRS được trình bày trong Bảng 2 và Bảng 3. Các kết quả được tính từ trung bình của 10-fold, bao gồm kích thước rút gọn (*size*), độ chính xác phân lớp (*acc*) và cặp tham số tối ưu (k, ε). Trong đó, (k, ε) là cặp tham số mang lại độ chính xác cao nhất trong toàn bộ dải giá trị khảo sát, như mô tả ở phần thiết lập.

Bảng 1. Mô tả chi tiết các tập dữ liệu.

STT	Tên tập dữ liệu	|C|	|U|	Lớp	STT	Tên tập dữ liệu	|C|	|U|	Lớp
1	meta-instance incremental	62	74	4	16	fri-c4-250-50	50	250	2
2	Meter A	36	87	2	17	Vertebral Column	6	310	2
3	Chemical Composion of Ceramic	17	88	2	18	leaf	15	340	30
4	Meter B	51	92	3	19	ionosphere	34	351	2
5	robot-failures-lp4	90	117	3	20	Pima Indians Diabetes	8	768	2
6	Algerian-Forest Fires-Bejaia	10	122	2	21	pc3	37	1563	2
7	kc1-binary	94	145	2	22	Steel plates fault	33	1941	2
8	wine	13	178	3	23	kc1	21	2109	2
9	Meter D	43	180	4	24	segment	19	2310	7
10	Meter C	43	181	4	25	fri-c4-100-100	100	100	2
11	triazines	60	186	2	26	spectrometer	100	531	48
12	wpbc	33	198	2	27	PersonGait DataSet	321	48	16
13	sonar	60	208	2	28	SRBCT	2308	83	4
14	seeds	7	210	3	29	warpAR10P	2400	130	10
15	glass	9	214	6	30	Lymphoma-3	4026	66	3

Kết quả cho thấy cả hai thuật toán đều giảm đáng kể số lượng thuộc tính so với tập dữ liệu ban đầu và tạo ra các tập rút gọn có số lượng thuộc tính tương đương nhau trên đa số tập dữ liệu. Về độ chính xác phân lớp trên bộ phân lớp Naive Bayes, SP-GWNRS vượt trội hơn GWNRS trên 24/30 tập dữ liệu. Đặc biệt, trên các tập fri-c4-100-100, glass, meta-instanceincremental, sonar, và warpAR10P, SP-GWNRS đạt độ chính xác cao hơn GWNRS từ 4% đến 12%. Điều này cho thấy, SP-GWNRS không chỉ có khả năng loại bỏ thuộc tính dư thừa mà còn duy trì hoặc nâng cao khả năng phân lớp. Các

trường hợp còn lại, hai thuật toán cho kết quả tương đương (ví dụ Algerian-Forest Fires-Bejaia, Chemical Composion of Ceramic, Meter D, pc3, hoặc steel-plates-fault) chứng tỏ rằng SP-GWNRS không làm suy giảm độ chính xác. Duy nhất tập dữ liệu SRBCT, thuật toán đề xuất cho kết quả thấp hơn GWNRS khoảng 1.11%, mức chênh lệch này không đáng kể.

Bảng 2. Kết quả so sánh hiệu quả hai thuật toán trên bộ phân lớp NaiveBayes.

Tập dữ liệu	**SP-GWNRS**			**GWNRS**		
	(k, ε)	***size***	***acc***	***(k, ε)***	***size***	***acc***
Algerian-Forest Fires-Bejaia	(1, 0.5)	1.1	**97.57±3.92**	(1, 0.5)	1.1	**97.57±3.92**
Chemical Composion of Ceramic	(1, 0.5)	1	**100±0**	(1, 0.5)	1	**100±0**
Lymphoma-3	(15, 0.7)	2.3	**98.33±5.27**	(5, 0.8)	2.2	96.9±6.55
Meter A	(3, 1.0)	3	**83.89±8.36**	(8, 0.5)	2.5	81.67±9.73
Meter B	(1, 0.5)	1.7	**100±0**	(1, 0.6)	1.9	**100±0**
Meter C	(24, 0.5)	5.5	**67.49±8.72**	(24, 0.5)	5.5	66.38±9.51
Meter D	(6, 0.5)	2.4	**85.56±10.54**	(16, 1.1)	2.2	85.55±8.76
PersonGaitDataSet	(9, 0.9)	7.1	**50.5±15.71**	(9, 0.7)	6.3	47±19.89
Pima Indians Diabetes	(69, 0.7)	3.8	**79.04±4.23**	(69, 0.7)	3.8	77.86±5
SRBCT	(10, 0.6)	4.3	95.28±6.11	(15, 1.4)	5.9	**96.39±5.83**
Vertebral Column	(2, 0.7)	5	**81.61±8.2**	(16, 0.5)	2.1	79.68±8.2
fri-c4-100-100	(5, 0.5)	3.5	**82±6.32**	(10, 1.2)	3.3	78±6.32
fri-c4-250-50	(4, 0.6)	3.5	**76.8±6.48**	(27, 0.6)	2.3	76±7.3
glass	(23, 1.0)	4.4	**61.69±7.76**	(9, 0.5)	3.9	49.48±11.7
ionosphere	(47, 0.6)	2.3	**83.47±5.2**	(47, 0.6)	2.3	82.61±5.96
kc1	(13, 0.7)	1.4	**85.35±2.9**	(13, 0.7)	1.4	84.87±2.92
kc1-binary	(9, 1.0)	2.9	**74.09±12.85**	(9, 1.4)	3.6	72.76±13.3
leaf	(6, 1.0)	9.2	**77.65±3.97**	(6, 0.5)	7.5	75.88±7.17
meta-instanceincremental	(14, 0.5)	3.2	**83.39±19.81**	(14, 1.2)	2.5	77.68±25.38
pc3	(8, 0.5)	1.1	**89.83±1.88**	(8, 0.5)	1.1	**89.83±1.88**
robot-failures-lp4	(3, 1.0)	3.5	**93.18±6.84**	(4, 1.1)	3.7	92.35±6.18
seeds	(1, 0.6)	4.8	**95.72±2.7**	(1, 0.5)	3.9	93.81±4.52
segment	(4, 0.9)	8.8	**88.92±2.2**	(4, 0.5)	7.5	87.75±2.1
sonar	(1, 1.0)	5.7	**82.31±8.32**	(20, 0.5)	3.7	74.07±11.01
spectrometer	(4, 0.5)	6.8	**53.3±6.68**	(9, 1.4)	7	50.46±5.24
steel-plates-fault	(5, 0.5)	7.9	**100±0**	(5, 0.5)	7.9	**100±0**

triazines	(5, 0.9)	4.9	**75.82±8.07**	(25, 1.3)	5.7	72.46±9.32
warpAR10P	(17, 0.7)	9.2	**76.92±14.95**	(12, 0.7)	8.5	71.54±13.59
wine	(11, 0.9)	4.7	**97.74±5.39**	(17, 1.0)	5	96.63±3.91
wpbc	(17, 0.6)	2.5	**79.84±8.76**	(7, 0.7)	1.8	78.87±9.31

Bảng 3. Kết quả so sánh hiệu quả hai thuật toán trên bộ phân lớp KNN.

Tập dữ liệu	**SP-GWNRS**			**GWNRS**		
	(k, ε)	***size***	***acc***	***(k, ε)***	***size***	***acc***
Algerian-Forest Fires-Bejaia	(9, 0.5)	1	**96.73±4.22**	(9, 0.5)	1	**96.73±4.22**
Chemical Composion of Ceramic	(1, 0.5)	1	**100±0**	(1, 0.5)	1	**100±0**
Lymphoma-3	(15, 0.7)	2.3	**98.33±5.27**	(15, 0.6)	2.7	96.9±6.55
Meter A	(14, 0.7)	2.7	**96.67±5.37**	(14, 0.6)	2.1	93.33±9.37
Meter B	(15, 0.5)	2	**100±0**	(15, 0.5)	2	**100±0**
Meter C	(3, 1.0)	3.3	98.33±3.75	(3, 1.4)	3.2	**98.89±2.34**
Meter D	(8, 0.6)	2.7	**88.33±8.47**	(8, 0.6)	2.7	**88.33±8.47**
PersonGaitDataSet	(9, 0.8)	6.5	**78.5±27.89**	(2, 0.8)	4	78±19.89
Pima Indians Diabetes	(62, 1.0)	4.6	**76.17±2.76**	(62, 1.3)	4	74.09±5.09
SRBCT	(7, 0.5)	4.3	**96.39±5.83**	(9, 1.3)	5.1	94.03±6.32
Vertebral Column	(2, 0.6)	4.8	**84.52±5.22**	(27, 0.7)	2	82.58±5.53
fri-c4-100-100	(7, 1.0)	4.3	**93±4.83**	(10, 0.5)	3.4	86±9.66
fri-c4-250-50	(22, 0.5)	3.3	**84.8±6.2**	(15, 0.7)	2.9	83.2±4.92
glass	(1, 0.8)	6.8	**77.99±6.44**	(7, 1.1)	5.2	73.31±8.47
ionosphere	(6, 0.8)	2.9	**83.75±5.59**	(6, 0.5)	2.7	82.32±5.55
kc1	(18, 0.9)	2.9	**55.58±22.16**	(18, 0.9)	2.9	55.39±21.99
kc1-binary	(9, 0.6)	2.8	**79.86±11.62**	(13, 1.3)	3.1	78.19±11.93
leaf	(33, 0.5)	8.4	**70.29±4.03**	(6, 0.5)	7.5	67.94±6.42
meta-instanceincremental	(5, 1.0)	3.4	**89.28±10.61**	(5, 1.0)	3.4	87.86±9.95
pc3	(8, 0.5)	1.1	**89.83±2.1**	(8, 0.5)	1.1	**89.83±2.1**
robot-failures-lp4	(3, 0.6)	3.3	**92.27±8.49**	(3, 1.3)	3.8	91.44±8.96
seeds	(20, 0.5)	2.4	**94.76±6.53**	(20, 0.5)	2.4	**94.76±6.53**
segment	(27, 0.6)	4.2	**97.53±1.02**	(27, 0.6)	4.2	97.4±1.08
sonar	(11, 0.9)	4.7	**83.19±7.14**	(20, 0.9)	5.2	79.88±10.29
spectrometer	(62, 0.5)	10	**58.96±7.38**	(9, 1.1)	6.6	56.87±3.86
steel-plates-fault	(5, 0.5)	7.9	**100±0**	(5, 0.5)	7.9	**100±0**
triazines	(25, 0.7)	5.8	**80±9.57**	(3, 0.6)	4.7	76.34±13.59
warpAR10P	(14, 0.7)	9.1	**82.31±10.91**	(12, 0.7)	8.5	79.23±10.91
wine	(1, 0.6)	3.8	**97.19±3.96**	(17, 1.0)	5	**97.19±3.96**
wpbc	(17, 0.6)	2.5	**76.76±10.33**	(21, 0.5)	2.2	73.68±12.21

Trên bộ phân lớp KNN, SP-GWNRS tiếp tục duy trì hoặc vượt trội độ chính xác so với GWNRS trên 29/30 tập dữ liệu. Chỉ duy nhất một trường hợp Meter C, GWNRS có độ chính xác cao hơn SP-GWNRS, tuy nhiên mức chênh lệch chỉ khoảng 0.56%, không

đáng kể. Đặc biệt, các tập Meter A, fri-c4-100-100, glass, sonar, triazines, warpAR10P và wpbc ghi nhận mức cải thiện rõ rệt từ 3 - 7%, trong khi kích thước rút gọn vẫn tương đối cạnh tranh so với GWNRS. Điều này chứng tỏ cơ chế cắt tỉa mềm động đã giúp thuật toán loại bỏ hiệu quả các thuộc tính không mang thông tin, từ đó nâng cao độ chính xác phân lớp.

Tổng thể, kết quả phân lớp khẳng định rằng SP-GWNRS không chỉ trả về tập rút gọn có kích thước tương đương thuật toán gốc, mà còn duy trì độ chính xác cao và ổn định trên nhiều loại dữ liệu. Cơ chế cắt tỉa mềm cho phép thuật toán thích ứng linh hoạt với đặc trưng dữ liệu khác nhau, giúp nâng cao tính tổng quát hóa, khả năng thích nghi và độ ổn định trong quá trình rút gọn thuộc tính.

5. KẾT LUẬN

Trong bài báo này, chúng tôi đã tập trung giải quyết bài toán rút gọn thuộc tính dựa trên cách tiếp cận GWNRS. Mặc dù thuật toán heuristic truyền thống có thể tìm được các tập rút gọn một cách hiệu quả, nhưng chi phí tính toán tăng khi áp dụng cho các tập dữ liệu quy mô lớn và có số chiều cao, do phải lặp lại việc đánh giá các thuộc tính không mang thông tin. Để khắc phục hạn chế này, chúng tôi đề xuất SP-GWNRS, một thuật toán heuristic mới dựa trên cơ chế cắt tỉa mềm. Mỗi thuộc tính được gán một bộ đếm (failure counter) và cập nhật động trong quá trình cắt tỉa và thiết lập lại, giúp thuật toán loại bỏ tạm thời những thuộc tính không quan trọng, đồng thời vẫn duy trì các thuộc tính tiềm năng có khả năng đóng góp vào kết quả cuối cùng. Chúng tôi đã phân tích độ phức tạp tính toán và thấy rằng thuật toán đạt được chi phí thực tế giảm đáng kể, được biểu diễn bởi: $O(\varrho \cdot |C|^2 \cdot |U|^2)$, trong đó ϱ là tỷ lệ cắt tỉa, thường nhỏ hơn nhiều so với 1 trong các tập dữ liệu có số chiều lớn. Thực nghiệm trên các hệ quyết định chuẩn cũng cho thấy SP-GWNRS tạo ra các tập rút gọn tương đương hoặc nhỏ hơn, trong khi độ chính xác phân lớp vượt trội so với thuật toán gốc.

Hướng nghiên cứu trong tương lai bao gồm: Mở rộng chiến lược cắt tỉa mềm cho các hệ quyết định động, kết hợp SP-GWNRS với các kỹ thuật tối ưu hóa metaheuristic, và ứng dụng thuật toán này vào các lĩnh vực thực tế có dữ liệu nhiều chiều như tin sinh học và khai phá văn bản.

LỜI CẢM ƠN

Nghiên cứu này được hỗ trợ bởi Đại học Huế cho đề tài DHH2024-01-217. Các tác giả cũng cảm ơn sự hỗ trợ của Trường Đại học Khoa học, Đại học Huế.

TÀI LIỆU THAM KHẢO

[1] Z. Pawlak (1982). Rough sets, Int. J. Comput. Inf. Sci. 11 341–356.

[2] Q. Hu, J. Liu, D. Yu (2008). Mixed feature selection based on granulation and approximation, Knowl.-Based Syst. 21 (4) 294–304.

[3] L. Sun, T. Yin, W. Ding, Y. Qian, J. Xu (2022). Feature selection with missing labels using multilabel fuzzy neighborhood rough sets and maximum relevance minimum redundancy, IEEE Trans. Fuzzy Syst. 30 (5) 1197–1211.

[4] S. An, X. Guo, C. Wang, G. Guo, J. Dai (2023). A soft neighborhood rough set model and its applications, Inf. Sci. 624 185–199.

[5] M. Hu, E.C. Tsang, Y. Guo, D. Chen, W. Xu (2021). A novel approach to attribute reduction based on weighted neighborhood rough sets, Knowl.-Based Syst. 220 106908.

[6] N.N. Thuy, S. Wongthanavasu (2024). Attribute reduction with fuzzy divergence-based weighted neighborhood rough sets, Int. J. Approx. Reason. 173 109256.

[7] N. Wang, E. Zhao (2024). A new method for feature selection based on weighted k-nearest neighborhood rough set, Expert Syst. Appl. 238 122324.

[8] C. Gao, J. Zhou, X. Wang and W. Pedrycz (2025). Granule Margin-Based Feature Selection in Weighted Neighborhood Systems, IEEE Transactions on Cybernetics, vol. 55, no. 5, pp. 2151-2164.

[9] Nguyen Ngoc Thuy, Tran Duy Anh, Le Manh Thanh (2025). Generalized weighted neighborhood rough sets, Information Sciences, Volume 707, 122020.

AN EFFICIENT FILTER-WRAPPER ATTRIBUTE REDUCTION ALGORITHM BASED ON SOFT PRUNING AND GENERALIZED WEIGHTED NEIGHBORHOOD ROUGH SETS

Tran Duy Anh, Le Manh Thanh, Doan Thi Hong Phuoc, Nguyen Ngoc Thuy*

Faculty of Information Technology, University of Sciences, Hue University

*Email: nnthuy.cs@hueuni.edu.vn

ABSTRACT

Attribute reduction is a central task in rough set theory, aiming to eliminate redundant attributes while preserving the classification ability of the decision system. Within the framework of generalized weighted neighborhood rough sets (GWNRS), the existing heuristic algorithm has shown some effectiveness but still incurs a high computational cost when applied to large-scale data. To overcome this limitation, we propose a new algorithm, SP-GWNRS, based on a soft pruning mechanism. Each attribute is associated with a counter that tracks the number of times it contributes negatively to the dependency degree, enabling the algorithm to adaptively balance exploration and pruning to reduce redundant evaluations. Furthermore, a wrapper phase is incorporated to select the optimal final reduct based on classification accuracy. Experimental results on benchmark decision systems demonstrate that SP-GWNRS achieves reductions of comparable size to those of the original heuristic algorithm, while outperforming it in classification accuracy.

Keywords: Rough set theory; generalized weighted neighborhood rough sets; attribute reduction; soft pruning; decision information systems.