Các công trình nghiên cứu, phát triển và ứng dụng
Công nghệ Thông tin và Truyền thông
https://www.ictmag.vn/cntt-tt
Tập 2022, Số 2, Tháng 12
Số đặc biệt VNICT 2022

MỤC LỤC

Kỹ thuật phát hiện nhanh va chạm của chất liệu vải tương tác trong môi trường thực tại ảo ..... ***Nghiêm Văn Hưng, Đặng Văn Đức, Nguyễn Văn Căn, Hoàng Việt Long, Trịnh Hiền Anh*** 49

Về một phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận topo mờ trực cảm ..... ***Trần Thanh Đại, Nguyễn Long Giang, Trần Thị Ngân, Hoàng Thị Minh Châu, Vũ Thu Uyên, Vương Trung Hiếu*** 57

Một kỹ thuật định vị trong nhà bằng WiFi hiệu quả sử dụng học máy kết hợp ..... ***Ngô Văn Bình, Vũ Văn Hiệu*** 65

Hệ gợi ý mua sắm dựa theo phiên làm việc với mô hình mạng học sâu đồ thị ..... ***Nguyễn Tuấn Khang, Nguyễn Tú Anh, Mai Thúy Nga, Nguyễn Hải An, Nguyễn Việt Anh*** 73

Một phương pháp xây dựng hệ thống nhận dạng hành vi của người trong thời gian thực dựa trên các thuật toán học máy ..... ***Nguyễn Quang Huy, Trần Đức Nghĩa, Đào Tô Hiệu, Trần Đức Tân, Vũ Thị Thương, Đỗ Xuân Trung*** 83

An Ontology-based Sentiment Analysis Approach to Discovering Hidden Affected Objects ..... ***Le Anh Phuong, Nguyen Minh Duc, Nguyen Dinh Hoa Cuong, Nguyen Thi Huong Giang*** 89

Using Graph Convolution Neural Networks for Solving Mixed integer Linear Programs ..... ***Nguyen Thi My Binh, Dao Hoang Long, Nguyen Van Thien, Pham Hong Thai*** 94

Các công trình nghiên cứu, phát triển và ứng dụng
Công nghệ Thông tin và Truyền thông
https://www.ictmag.vn/cntt-tt
Tập 2022, Số 2, Tháng 12
Số đặc biệt VNICT 2022

TABLE OF CONTENT

Fast Collision Detection Technique of Interactive Fabrics in Virtual Reality Environment ..... ***Nghiem Van Hung, Đang Van Đuc, Nguyen Van Can, Hoang Viet Long, Trinh Hien Anh*** 49

Intuitionistic Fuzzy Topology-Based Attribute Reduction on the Decision Table ..... ***Tran Thanh Đai, Nguyen Long Giang, Tran Thi Ngan, Hoang Thi Minh Chau, Vu Thu Uyen, Vuong Trung Hieu*** 57

An Efficient WiFi Indoor Positioning Technique using Combined Machine Learning ..... ***Ngo Van Binh, Vu Van Hieu*** 65

Session-based Recommendation using Graph Neural Network ..... ***Nguyen Tuan Khang, Nguyen Tu Anh, Mai Thuy Nga, Nguyen Hai An, Nguyen Viet Anh*** 73

A Method of Building a Real-time Human Behavior Recognition System based on Machine Learning Algorithms ..... ***Nguyen Quang Huy, Tran Đuc Nghia, Đao To Hieu, Tran Đuc Tan, Vu Thi Thuong, Đo Xuan Trung*** 83

An Ontology-based Sentiment Analysis Approach to Discovering Hidden Affected Objects ..... ***Le Anh Phuong, Nguyen Minh Duc, Nguyen Dinh Hoa Cuong, Nguyen Thi Huong Giang*** 89

Using Graph Convolution Neural Networks for Solving Mixed integer Linear Programs ..... ***Nguyen Thi My Binh, Dao Hoang Long, Nguyen Van Thien, Pham Hong Thai*** 94

Kỹ thuật phát hiện nhanh va chạm của chất liệu vải tương tác trong môi trường thực tại ảo

Nghiêm Văn Hưng[1,2], Đặng Văn Đức[3], Nguyễn Văn Căn[2], Hoàng Việt Long[2], Trịnh Hiền Anh[3]

[1] Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội, Việt Nam

[2] Trường Đại học Kỹ thuật - Hậu cần Công an nhân dân, Bộ Công an, Bắc Ninh, Việt Nam

[3] Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội, Việt Nam

Tác giả liên hệ: Nghiêm Văn Hưng, nghiemvanhung1985@gmail.com

Ngày nhận bài: 29/07/2022, ngày sửa chữa: 12/11/2022, ngày duyệt đăng: 30/11/2022

Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1128

Tóm tắt: Va chạm là một trong những vấn đề cơ bản cần phải xử lý trong các hệ thống đồ họa máy tính và tương tác thực tại ảo. Phát hiện va chạm là khâu mở đầu, mang tính chất quyết định đến các tác vụ xử lý va chạm của hệ thống. Trong bài báo này, chúng tôi trình bày một số cải tiến để phát hiện nhanh va chạm của mô hình vải được biểu diễn dưới dạng mặt tam giác. Sự tương tác của vải với các vật thể khác và với chính nó như gập, cuộn làm cho quá trình phát hiện va chạm trở nên phức tạp hơn. Kỹ thuật đề xuất sử dụng thuật toán lọc pha rộng và lọc pha hẹp được thử nghiệm và cho kết quả tốt trên ba bộ dữ liệu mô hình vải khác nhau trong thư viện GAMMA, tốc độ phát hiện va chạm trung bình nhanh gấp khoảng 2.3 lần so với phương pháp của Tang và cộng sự, nhanh gấp khoảng 1.34 lần so với phương pháp của Zhang và cộng sự.

Từ khóa: *Phát hiện va chạm, mô phỏng, mô hình 3D.*

Title: **Fast Collision Detection Technique of Interactive Fabrics in Virtual Reality Environment**

Abstract: Collisions are one of the fundamental problems that need to be dealt with in computer graphics and virtual reality systems. Collision detection is the first step, which is decisive for the system's collision handling tasks. In this paper, we present some improvements for fast collision detection of fabric model represented as triangle mesh. The interaction of the fabric with other objects and with itself, such as folding and rolling, makes the collision detection process more complicated. The proposed technique using the broad-phase filter algorithm and the narrow-phase filter algorithm is tested and gives good results on three different fabric model datasets in GAMMA library, the average collision detection speed is about 2.3 times faster times compared to the method of Tang et al., about 1.34 times faster than the method of Zhang et al.

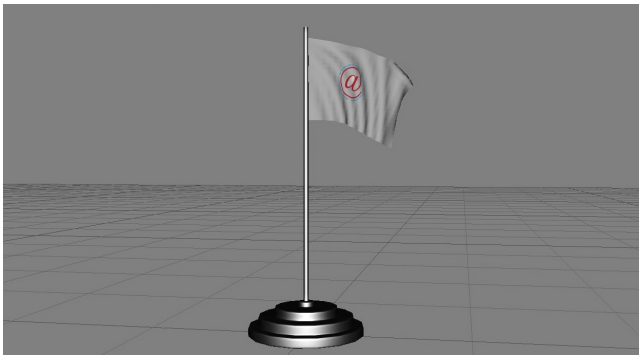
Keywords: *Collision detection, simulation, 3D modeling.*

I. GIỚI THIỆU

Phát hiện va chạm là một trong những tác vụ cơ bản của các hệ thống mô phỏng thực tại ảo, đồ họa máy tính, điều khiển robotics, games,... Các đối tượng 3D trong hệ thống mô phỏng có thể là các mô hình vật thể rắn hoặc mô hình vật thể biến dạng, trong đó vải là một trong những mô hình vật thể biến dạng điển hình. Bài báo này tập trung vào vấn đề phát hiện va chạm của mô hình vải được biểu diễn dưới dạng mặt tam giác. Hình 1 minh họa quá trình va chạm của vải trên một lá cờ có hình biểu trưng của Hội thảo VNICT đang tung bay trong gió.

Kế thừa và phát triển từ các phương pháp của Tang và cộng sự [10], Zhang và cộng sự [20] về phát hiện va chạm trong các mô hình vật thể biến dạng, kỹ thuật đề xuất trong bài báo này cho phép tăng nhanh tốc độ phát hiện va chạm trong các mô hình vải khác nhau thuộc bộ dữ liệu GAMMA (Geometric Algorithms for Modeling, Motion, and Animation) thuộc Trường Đại học Maryland, Hoa Kỳ.

Phần còn lại của bài báo được cấu trúc như sau: Mục II trình bày về các nghiên cứu liên quan. Mục III trình bày kỹ thuật đề xuất. Kết quả thử nghiệm được trình bày trong mục IV. Mục V phân tích đánh giá, mục VI là kết luận và định hướng nghiên cứu tiếp theo.



Hình 1. Mô phỏng một lá cờ đang tung bay theo gió, trong đó có các phần vải va chạm và tự va chạm

II. CÁC NGHIÊN CỨU LIÊN QUAN

Đã có nhiều công bố cải tiến kỹ thuật phát hiện va chạm trong các mô hình vật thể biến dạng [3, 4, 15, 16], hầu hết đều dựa trên cấu trúc phân hệ vùng bao (Bounding Volume Hierarchies - BVH). Đối với các mô hình vật thể biến dạng như vải thì chi phí duyệt và tái cấu trúc BVH là rất lớn, vấn đề đặt ra là cần thiết kế thuật toán có thể hoạt động tốt trên các cấu hình biến dạng khác nhau. Curtis và cộng sự [4] đề xuất sử dụng tam giác đại diện để loại bỏ các phép kiểm tra lặp, mặc dù có tăng hiệu quả phát hiện va chạm, nhưng do tính ngẫu nhiên của thuật toán phân bổ của nó, nên khó tích hợp với các thuật toán lọc khác. Sau đó, Tang và cộng sự [10] đã thực hiện một số cải tiến trên phương pháp của Curtis và cộng sự [4] để giảm hơn nữa số lượng các phép kiểm tra cơ bản, có khả năng mở rộng tốt, chi phí lưu trữ thấp. Trong khi đó, Brochu [1], Tang [13] và Wang [14] đưa ra các yêu cầu cao hơn về độ chính xác tính toán của hệ thống.

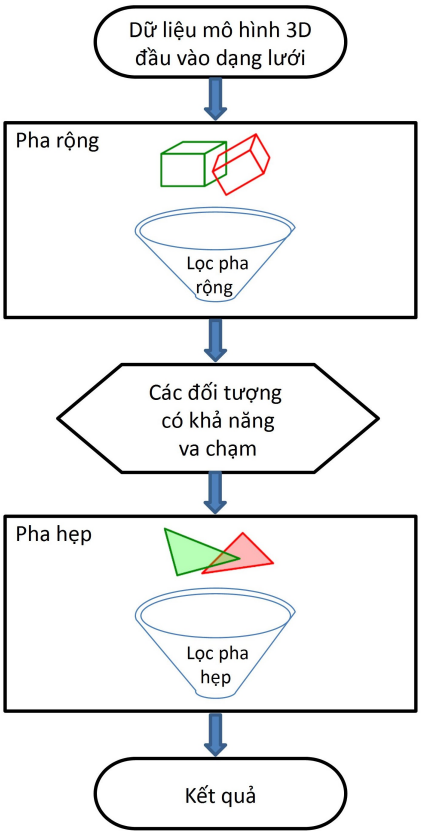
Các hệ thống mô phỏng thực tại ảo thường xuyên phải thực thi các tác vụ phát hiện va chạm. Quá trình phát hiện va chạm chiếm rất nhiều thời gian trong các mô phỏng có sự chuyển động, đồng thời cũng là phần khó và phức tạp về tính toán, thậm chí có thể gây ra tình trạng nghẽn nút cổ chai [5, 9]. Nhìn chung, có hai hướng giải pháp để tăng tốc phát hiện va chạm: hướng thứ nhất là sử dụng cấu trúc phân hệ vùng bao động [15, 16] và hướng thứ hai là sử dụng thuật toán lọc[5, 12, 19, 20]. Tuy nhiên, trong những mô phỏng va chạm mà các mô hình có mức độ biến dạng rất lớn thì phương pháp sử dụng thuật toán lọc trong công bố của Du [5] và Tang [12] tỏ ra kém hiệu quả vì không xử lý được trường hợp lưới tam giác bị lật.

III. KỸ THUẬT ĐỀ XUẤT

1. A. Mô tả kỹ thuật

a) Mô tả chung

Nghiên cứu của chúng tôi kế thừa và phát triển trên cơ sở giải pháp lọc của Tang [10] và Zhang [20] vì tính khả thi và phù hợp đối với vật thể chất liệu vải. Kết quả thử nghiệm cho thấy kỹ thuật đề xuất hiệu quả ngay cả khi mô hình vải bị biến dạng rất lớn. Quá trình phát hiện va chạm được chia thành hai pha là pha rộng và pha hẹp như minh họa trong Hình 2. Trong đó, pha rộng là quá trình kiểm tra va chạm giữa các hộp bao, pha hẹp là quá trình kiểm tra va chạm giữa các mắt lưới tam giác. Pha rộng loại trừ các cặp đối tượng không thể giao nhau, kết quả của pha này là chọn ra được các đối tượng có khả năng va chạm. Đầu ra của pha rộng là đầu vào của pha hẹp. Pha hẹp tính toán để xác định chính xác những đối tượng thực sự va chạm. Chúng tôi đề xuất một kỹ thuật tăng tốc phát hiện va chạm cho mô hình vải dựa trên các thuật toán lọc pha rộng và lọc pha hẹp minh họa trong Hình 2. Kỹ thuật đề xuất loại bỏ các cặp đối tượng không có khả năng xảy ra va chạm và tăng tốc phát hiện va chạm.



Hình 2. Sơ đồ tổng quát kỹ thuật đề xuất

Phân hệ vùng bao BVH là một cấu trúc dữ liệu dạng cây được tạo thành trên cơ sở phân tích các đối tượng cần được kiểm tra va chạm. BVH đóng vai trò quan trọng trong việc biểu diễn các vật thể và xử lý phát hiện va chạm.

Cấu trúc BVTT (Bounding Volume Traversal Tree) biểu diễn hệ thống phân cấp của các phép kiểm tra chồng lấp (overlap tests) được thực hiện trong quá trình duyệt BVH. Mỗi nút trong BVTT đại diện cho một phép kiểm tra chồng lấp duy nhất giữa một cặp khối bao.

Công đoạn đầu tiên là khởi tạo mô hình ba chiều đầu vào và sử dụng phương pháp ICCD của Tang [10] để đánh dấu nhằm đảm bảo rằng việc kiểm tra cặp đối tượng cơ sở chỉ được thực hiện một lần. Sau đó, các khối bao BV (Bounding Volume) được xây dựng. Tùy theo các BV có chồng lấp nhau hay không, có thể loại trừ một số cặp tam giác và các cặp đối tượng cơ sở không có khả năng va chạm.

Chuyển sang công đoạn kế tiếp, phương pháp lọc dựa trên tính chất hình học của Tang [12] được sử dụng để xác định tính đồng phẳng, và sau đó phương pháp lọc dựa vào phép toán đại số của Zhang [20] được sử dụng để xác định sự tồn tại nghiệm của phương trình khoảng cách.

b) Lọc pha rộng

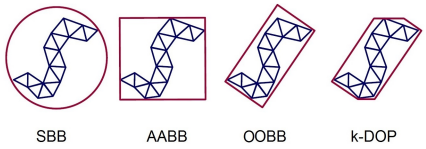
Việc phát hiện va chạm trên các mô hình vải tập trung vào hai vấn đề chủ yếu là các phép kiểm tra cơ sở và tự va chạm.

Các phép kiểm tra cơ sở: Thực hiện 15 phép kiểm tra cơ sở giữa các mặt (Face), cạnh (Edge) và đỉnh (Vertex) của hai tam giác, gồm 6 phép kiểm tra đỉnh-mặt (Vertex-Face ghi tắt là VF) và 9 phép kiểm tra cạnh-cạnh (Edge-Edge ghi tắt là EE).

Tự va chạm: Do chuyển động biến dạng và thay đổi hình dạng có thể dẫn đến tự va chạm.

Mỗi vật thể ba chiều (3D) được tạo thành từ nhiều mặt, do đó chi phí để kiểm tra va chạm là rất cao. Hầu hết các hệ thống thực tế ảo sử dụng phương pháp phát hiện va chạm dựa trên các khối bao. Khối bao là một không gian hình học khép kín bao bọc hoàn toàn đối tượng. Hình 3 cho thấy một số loại khối bao được sử dụng trong các kỹ thuật phát hiện va chạm: khối bao căn chỉnh theo trục AABB (Axis Aligned Bounding Box), khối bao hình cầu SBB (Sphere Bounding Box), khối bao xác định theo hướng đối tượng OOBB (Object-Oriented Bouding Box) và khối bao đa diện rời rạc có hướng k-DOP (k-Discrete Oriented Polytopes). Bảng I thể hiện kết quả so sánh các loại khối bao phổ biến theo các tiêu chí đánh giá khác nhau.

Có bốn tiêu chí đánh giá trong bảng trên và mỗi tiêu chí đánh giá được chia làm bốn mức độ được đánh số từ 1 đến 4 (quy ước giá trị số 1 là tốt nhất, giá trị số càng cao thì mức độ đáp ứng càng thấp). Việc chọn loại khối bao được xác định bởi nhiều yếu tố như: chi phí xây dựng khối bao, chi phí cập nhật lại khối bao khi đối tượng di chuyển hoặc thay đổi hình dạng/kích thước, chi phí xác định điểm



Hình 3. Một số loại khối bao được sử dụng trong các kỹ thuật phát hiện va chạm

va chạm và mức độ chính xác của phương pháp phát hiện va chạm,... Trong bài báo này, chúng tôi lựa chọn sử dụng khối bao k-DOP để thỏa hiệp giữa các tiêu chí đánh giá và thuận lợi cho việc tính toán phát hiện va chạm của các mô hình vải. Khối bao k-DOP cũng được sử dụng trong các nghiên cứu của Curtis [4], Du [5, 6], Tang [10, 12, 13] và Zhang [19, 20].

Các đối tượng được biểu diễn bằng lưới tam giác. Bề mặt lưới cung cấp rất nhiều thông tin hữu ích có thể được sử dụng để tăng tốc phát hiện va chạm. Nếu muốn kiểm tra xem hai tam giác t_a và t_b có cắt nhau trong thời gian ΔT hay không, các phương pháp phát hiện va chạm truyền thống cần tiến hành giải 15 phương trình bậc ba, trong đó gồm 6 phép kiểm tra VF và 9 phép kiểm tra EE.

Chúng tôi cho rằng không cần thiết phải sử dụng đến 15 phép kiểm tra, có thể cải tiến và thay thế bằng một chiến lược mới để giảm số lượng phép kiểm tra. Với t_a và t_b là kề nhau thì chỉ cần đến 4 phép kiểm tra là: (v_a,t_b),(v_b,t_a),(e_a^1,e_b^1) và (e_a^2,e_b^2). Bởi vì việc phát hiện va chạm giữa các tam giác liền kề dễ dàng hơn so với các tam giác không liền kề, toàn bộ quá trình phát hiện va chạm được chia thành hai bước. Bước đầu tiên kiểm tra các tam giác không liền kề và bước thứ hai kiểm tra các tam giác liền kề. Vì vậy, bước thứ hai sẽ tận dụng lợi thế này. Trong quá trình kiểm tra không liền kề, (t_g,t_b) và (v_a,t_b) được kiểm tra đồng thời. Tương tự, (t_a,t_h) và (v_b,t_a) được kiểm tra đồng thời, (t_e,t_f) và (e_a^1,e_b^1) được kiểm tra đồng thời, (t_c,t_d) và (e_a^2,e_b^2) được kiểm tra đồng thời. Vì vậy, trong tình huống này nếu phép kiểm tra các tam giác không liền kề được thực hiện trước thì có thể tránh được tất cả các phép kiểm tra các tam giác liền kề. Để cải thiện hiệu quả lọc, BVH được xây dựng từ ba dạng BV: BV đỉnh (V-BV), BV cạnh (E-BV) và BV mặt (F-BV) [4]. Khi BV của hai tam giác chồng lên nhau, về mặt lý thuyết thì cần tiến hành 15 phép kiểm tra. Nhưng trên thực tế, nếu chúng tôi sử dụng các E-BV thì chỉ cần 2 phép kiểm tra.

Tại thời điểm t_0 thì tam giác ở vị trí thấp hơn và tại thời điểm t_1 thì tam giác chuyển động đến vị trí cao hơn. Về mặt lý thuyết thì có thể sử dụng một BV lớn để bao phủ cả hai vị trí, dẫn đến việc phải tiến hành giải nhiều phương trình bậc ba để kiểm tra xem trong thời gian ΔT, đỉnh có chạm vào tam giác hay không. Trong tình huống này, kỹ thuật lọc không thâm nhập là một cách hiệu quả

Bảng I
SO SÁNH CÁC LOẠI KHỐI BAO

Loại khối bao	Tính đơn giản	Sử dụng bộ nhớ hiệu quả	Mức độ bó sát (tight-fighting)	Phù hợp cho mô hình vải
SBB	2	1	4	3
AABB	1	2	3	1
OOBB	4	3	2	4
k-DOP	3	4	1	2

và dễ dàng hơn nhiều để thực hiện phép kiểm tra này. Giả sử pháp tuyến của tam giác thay đổi từ $\vec{n_0}$ thành $\vec{n_1}$, và $\vec{n_t}$ là pháp tuyến tại thời điểm bất kỳ. Chiếu đỉnh a và p lên pháp tuyến $\vec{n_t}$, nếu trong thời gian ΔT khoảng cách chiếu luôn khác không thì đỉnh và tam giác không thể tiếp xúc nhau. Để xác định được khoảng cách này cần sử dụng đến các phép tính toán, tuy nhiên việc tính toán đơn giản hơn nhiều so với việc giải các phương trình bậc ba.

c) Lọc pha hẹp

Giả thiết rằng các đỉnh của tam giác chuyển động với vận tốc không đổi trong khoảng thời gian $t \in [0, 1]$. Phát hiện va chạm liên tục giữa hai tam giác chuyển động/biến dạng bao gồm một tập các phép kiểm tra VF hoặc EE. Cho bốn điểm $x_i\vec{}(t)$ và vận tốc không đổi của chúng $v_i\vec{}(t)$ với i=(1,2,3,4), hàm khoảng cách d(t) giữa đỉnh $\vec{x_4}$ và tam giác $\vec{x_1}\vec{x_2}\vec{x_3}$ hoặc giữa cạnh $\vec{x_1}\vec{x_2}$ với cạnh $\vec{x_3}\vec{x_4}$ được xác định như sau:

$$d(t) = [\vec{x_2}(t) - \vec{x_1}(t)] \times [\vec{x_3}(t) - \vec{x_1}(t)] \times [\vec{x_4}(t) - \vec{x_1}(t)] \quad (1)$$

trong đó $t \in [0, 1]$, $v_i\vec{}(t)$=$x_i\vec{}(1)$-$x_i\vec{}(0)$, $x_i\vec{}(t)$=$x_i\vec{}(0)$+$t\vec{v_i}$, ký hiệu toán tử "×" biểu diễn phép tích chéo, ký hiệu toán tử " ." biểu diễn phép tích vô hướng. Để đơn giản, ta đặt $\vec{x_i}$=$x_i\vec{}(0)$, do đó công thức (1) có thể biểu diễn lại dưới dạng:

$$d(t) = [\vec{x_{2_1}} + t\vec{v_{2_1}}] \times [\vec{x_{3_1}} + t\vec{v_{3_1}}] \times [\vec{x_{4_1}} + t\vec{v_{4_1}}] \quad (2)$$

trong đó ta sử dụng cách viết ngắn gọn $\vec{x_{i_j}}$ và $\vec{v_{i_j}}$ để biểu thị $\vec{x_i}$-$\vec{v_i}$ và $\vec{v_i}$-$\vec{v_j}$ tương ứng. Sau khi sắp xếp lại các thành phần thu được d(t) dưới dạng một đa thức bậc ba như sau:

$$d(t) = a_3t^3 + a_2t^2 + a_1t + a_0 \quad (3)$$

trong đó: $a_3 = \vec{x_{2_1}} \times \vec{x_{3_1}}$.

Bốn điểm đồng phẳng (tức là d(t)=0) là điều kiện cần thiết để xảy ra va chạm giữa hai tam giác. Do đó, một phương trình bậc ba có thể được suy ra. Nghiệm $t \in [0, 1]$ của phương trình d(t)=0 sẽ là thời gian tiếp xúc của hai tam giác. Trong thuật toán, va chạm được phát hiện khi khoảng cách nhỏ hơn ngưỡng cho trước. Nếu không tồn tại $t \in [0, 1]$ thì không có va chạm giữa hai tam giác trong khoảng thời gian [0,1].

Chúng tôi kết hợp thuật toán lọc dựa trên các phép toán đại số và thuật toán lọc dựa trên các đặc điểm hình học [12] liên quan đến tính đồng phẳng của các cặp VF, EE.

Ý tưởng chính là xác định xem có tồn tại bất kỳ nghiệm nào trong khoảng thời gian [0,1] hay không bằng những cách đơn giản và chi phí thấp. Nếu sự không tồn tại của nghiệm nguyên trong khoảng thời gian $t \in [0, 1]$ được xác định sớm hơn thì có thể tránh được việc phải giải các phương trình bậc ba với chi phí tính toán khá cao. Cụ thể, chúng tôi áp dụng quy tắc dấu Descartes và định lý Vincent để phân ly nghiệm.

Định lý 1. (Quy tắc dấu Descartes) Gọi f(t) là một đa thức theo lũy thừa giảm dần của t. Biến đổi dấu xảy ra bất cứ khi nào các hệ số khác không liền kề là khác dấu. Ký hiệu VAR(f(t)) là số cặp hệ số khác không liền kề có dấu ngược nhau. Trên thực tế, ta chỉ quan tâm đến đoạn [0,1]. Định lý Vincent là kết quả của quy tắc Descartes về dấu khi ánh xạ t từ (0,∞) đến (0,1) với ánh xạ $t \hookrightarrow 1/(t+1)$.

Định lý 2. (Định lý Vincent) Xét hàm $g(t) = (t+1)^n f(1/(t+1))$, nếu VAR(g(t))=0, với n là bậc của f(t) thì f(t) không có nghiệm nào trên [0,1]. Số nghiệm thực dương của g(t) bằng số nghiệm thực của f(t) trên [0,1]. Về hàm khoảng cách d(t) trong công thức (3) là f(t), theo Định lý 2, g(t) được biểu diễn dưới dạng công thức (4). Áp dụng quy tắc dấu Descartes cho g(t), ta có kết luận như trong Bảng II dưới đây:

Bảng II
SỰ TỒN TẠI CỦA NGHIỆM

VAR(g(t))	Số nghiệm
0	0
1	1
2	2 hoặc 0
3	3 hoặc 1

Từ Bảng II có thể thấy rằng sử dụng Định lý 2 cho hàm khoảng cách d(t) có thể nhanh chóng xác định sự tồn tại của nghiệm của phương trình khoảng cách. Khi VAR(g(t)) bằng 1 hoặc 3, có thể đánh giá rằng hàm khoảng cách có nghiệm. Trong trường hợp này, phương trình khoảng cách cần được tính toán chính xác. Nếu VAR(g(t)) bằng 0 thì phương trình không có nghiệm nguyên trên [0,1]. VAR(g(t)) bằng 2 thì

sự tồn tại của các nghiệm là không chắc chắn. Lúc này, nghiệm có thể không tồn tại hoặc có thể có 2 nghiệm. Trong trường hợp này, g(t) cần được phân tích thêm. Đầu tiên, phân tích g(t) thành $g_1(t)$ và $g_2(t)$ như trong công thức dưới đây. Hệ số b_i với i=(1,2,3,4) trong công thức (4) có thể được tính bằng a_i với i=(1,2,3,4) trong công thức (3).

$$g(t) = (t+1)^n f(1/t+1) = b_3t^3 + b_2t^2 + b_1t + b_0 \quad (4)$$

trong đó $b_3 = a_0$
$b_2 = a_1 + 3a_0$
$b_1 = a_2 + 2a_1 + 3a_0$
$b_0 = a_3 + a_2 + a_1 + a_0$
và

$$\begin{aligned} g_1(t) &= g(t+1) \\ &= b_3t^3 + (b_2+3b_3)t^2 + (b_1+2b_2+3b_3)t \\ &+ (b_0+b_1+b_2+b_3) \end{aligned} \quad (5)$$

và

$$\begin{aligned} g_2(t) &= (t+1)^3 g(1/t+1) \\ &= b_0t^3 + (b_1+3b_0)t^2 + (b_2+2b_1+3b_0)t \\ &+ (b_0+b_1+b_2+b_3) \end{aligned} \quad (6)$$

Nếu $VAR(g_1(t)) = 0$ thì f(t) (tức là d(t)) không có nghiệm trên [0,1/2].

Nếu $VAR(g_2(t)) = 0$ thì f(t) không có nghiệm nào trên [1/2,1].

Nếu VAR(g(t))=2 và $VAR(g_1(t)) = 0$ và $VAR(g_2(t)) = 0$ thì có thể kết luận rằng d(t) không có nghiệm nào trên [0,1]. Do đó có thể tránh được việc giải phương trình bậc ba.

2. Thuật toán

a) Thuật toán pha rộng

Pha rộng loại trừ các cặp đối tượng không có khả năng xảy ra va chạm, kết quả của pha này là chọn lọc ra được các đối tượng có khả năng va chạm (nếu có). Thuật toán Algorithm1 thực hiện tính toán tại mỗi frame để trả về các cặp đối tượng có khả năng xảy ra va chạm.

b) Thuật toán pha hẹp

Pha hẹp tính toán để xác định chính xác có xảy ra sự va chạm hay không. Việc đầu tiên là thuật toán Algorithm2 tiến hành kiểm tra tính đồng phẳng của các cặp VF (hoặc EE). Sau đó áp dụng phương pháp đại số để xác định sự tồn tại hay không tồn tại nghiệm của phương trình khoảng cách và đưa ra các kết luận tương ứng.

Thuật toán 1: Algorithm1 //Lọc pha rộng

```
Dữ liệu vào: Mô hình 3D được biểu diễn dạng lưới tam giác.
Dữ liệu ra: Các cặp đối tượng có khả năng xảy ra va chạm.
begin
    Tạo BV;
    Thiết lập cấu trúc phân cấp BVH theo chiều từ trên xuống;
    Thiết lập cấu trúc cây BVTT dựa vào phép duyệt cấu trúc phân cấp BVH;
    Kiểm tra sự chồng lấp, loại bỏ các cặp cặp đối tượng không có khả năng xảy ra va chạm tương ứng là với nút lá của cây BVTT; if không chồng lấp then
    end
    exit;
    (có khả năng xảy ra va chạm) thì chuyển sang thực hiện Algorithm2;
end
```

Thuật toán 2: Algorithm2 //Lọc pha hẹp

```
Dữ liệu vào: Tọa độ 04 đỉnh của cặp VF hoặc cặp EE cần kiểm tra va chạm.
Dữ liệu ra: Kết luận va chạm (thông tin điểm tiếp xúc, vectơ pháp tuyến của mặt tiếp xúc,...) hoặc kết luận không va chạm.
begin
    Kiểm tra và thực hiện lọc dựa vào các đặc điểm hình học (tính đồng phẳng của cặp VF hoặc cặp EE);
    Tính VAR(f(x)) để phân tích sự tồn nghiệm trong [0, 1];
    Áp dụng quy tắc dấu Descartes và định lý Vincent;
end
```

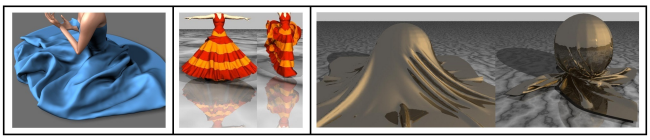
IV. THỬ NGHIỆM VÀ KẾT QUẢ

1. Môi trường thử nghiệm

Môi trường thử nghiệm là máy tính CPU Intel (R) Core (TM) i7-7820HQ @ 2,90 GHz, sử dụng ngôn ngữ C++ trên Windows và các tập dữ liệu GAMMA (Đại học Maryland).

2. Dữ liệu

Ba bộ dữ liệu được sử dụng để đánh giá hiệu suất của các thuật toán bao gồm:



(a) (b) (c)

Hình 4. Minh họa trên các bộ dữ liệu trong thư viện mở GAMMA: (a) Bộ dữ liệu Princess, (b) Bộ dữ liệu Flamenco, (c) Bộ dữ liệu Cloth-ball

Bộ dữ liệu Princess (Hình 4a): Mô phỏng một cô gái mặc chiếc váy dài thực hiện động tác múa nhẹ nhàng. Mô hình gồm 40000 tam giác.

Bộ dữ liệu Flamenco (Hình 4b): Mô phỏng một cô gái mặc chiếc váy xòe có 07 lớp vải thực hiện các động tác múa theo vũ điệu Flamenco. Bộ dữ liệu này có số lượng vải tự va chạm rất lớn. Mô hình gồm 49000 tam giác.

Bộ dữ liệu Cloth-ball (Hình 4c): Mô phỏng một mảnh vải (gồm 92230 tam giác) phủ trên đỉnh của quả cầu và khi quả cầu cuộn tròn cuốn theo mảnh vải dẫn đến có nhiều phần vải tự va chạm.

Bảng III
THÔNG TIN VỀ BỘ DỮ LIỆU

Bộ dữ liệu	Số tam giác	Số đỉnh	Số cạnh	Số frames
Pincess	40000	20000	60000	464
Flamenco	49000	26000	75000	352
Cloth-ball	92230	46598	138825	94

Mỗi bộ dữ liệu có nhiều frames được mô tả thông tin chi tiết trong BẢNG 3. Từ dữ liệu về các mô hình, có thể thấy rằng nếu tất cả các đỉnh, mặt, cạnh được kiểm tra theo từng cặp theo cách "vét cạn" mà không sử dụng các thuật toán lọc thì lượng tính toán là rất lớn.

a) Kết quả

Kết quả trong BẢNG 4 cho thấy hiệu quả của phương pháp đề xuất được sử dụng trong bài báo này. So sánh cho thấy Flamenco có hiệu quả loại bỏ ở mức cao. Các thuật toán lọc hai pha có thể phát huy đặc tính lọc tốt khi mô hình vải có mức độ biến dạng rất lớn.

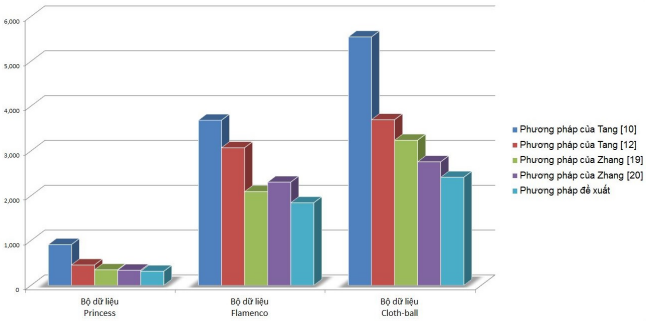
V. SO SÁNH, PHÂN TÍCH VÀ ĐÁNH GIÁ

1. So sánh và phân tích

Chúng tôi so sánh kết quả thử nghiệm phương pháp đề xuất với các phương pháp của Tang [10, 12] và Zhang [19, 20].

Phương pháp của chúng tôi có hai ưu điểm nổi bật: Thứ nhất là thời gian thực hiện kiểm tra chồng lấp khối bao và duyệt BVH được giảm đáng kể đối với các mô hình vải trong bộ dữ liệu thử nghiệm vì loại bỏ sớm các đối tượng không có khả năng va chạm; thứ hai là chiến lược lọc hai pha hiệu quả giảm được dương tính giả (False positive-FP) và tăng tốc phát hiện va chạm.

Biểu đồ trong Hình 5 so sánh cho thấy phương pháp đề xuất có kết quả chạy nhanh hơn. Lý do là chúng tôi sử dụng kết hợp cả yếu tố hình học và đại số để loại bỏ các cặp VF và các cặp EE không va chạm, tiến hành đánh giá



Hình 5. Một số loại khối bao được sử dụng trong các kỹ thuật phát hiện va chạm

sự tồn tại nghiệm của phương trình khoảng cách, tiết kiệm một phần thời gian cho các thao tác giải phương trình, giảm số lượng phép toán phức tạp.

a) Hạn chế

Phương pháp trong bài báo chỉ áp dụng cho trường hợp cấu trúc lưới tương đối ổn định, nếu mô hình bị rách hoặc vỡ trong quá trình va chạm thì phương pháp chưa xử lý hiệu quả. Ngoài ra, năng lực tính toán của GPU đã được tăng lên đáng kể trong những năm gần đây, để có thể phát hiện va chạm theo thời gian thực thì phương pháp đề xuất cần mở rộng cách tiếp cận với GPU.

VI. KẾT LUẬN, ĐỊNH HƯỚNG NGHIÊN CỨU PHÁT TRIỂN

Trong bài báo này, chúng tôi trình bày một số cải tiến để phát hiện nhanh va chạm của mô hình vải được biểu diễn dưới dạng mặt tam giác. Phương pháp đề xuất sử dụng thuật toán lọc pha rộng và lọc pha hẹp được thử nghiệm và cho kết quả tốt trên ba bộ dữ liệu mô hình vải khác nhau trong thư viện GAMMA, tốc độ phát hiện va chạm nhanh gấp khoảng 2.3 lần so với phương pháp của Tang và cộng sự, nhanh gấp khoảng 1.34 lần so với phương pháp của Zhang và cộng sư. Trong tương lai, chúng tôi dự kiến nghiên cứu xử lý va chạm với các cấu trúc lưới mô hình có thể bị cắt hoặc rách, vỡ,... và mở rộng cách tiếp cận này với GPU để cải thiện và tăng tốc thuật toán hơn nữa.

VII. LỜI CẢM ƠN

Chúng tôi xin chân thành cảm ơn các tác giả của ba bộ dữ liệu mô hình vải khác nhau trong thư viện GAMMA, các tác giả Tang, Zhang và cộng sự.

TÀI LIỆU THAM KHẢO

[1] Brochu T., Edwards E., Bridson R., "Efficient geometrically exact continuous collision detection", *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 1-7, 2012.

Bảng IV
SO SÁNH THỜI GIAN PHÁT HIỆN VA CHẠM TRUNG BÌNH CỦA CÁC PHƯƠNG PHÁP

Bộ dữ liệu	**Phương pháp của Tang [10]**	**Phương pháp của Tang [12]**	**Phương pháp của Zhang [19]**	**Phương pháp của Zhang [20]**	**Phương pháp đề xuất**
Princess	910	455	347	337	313
Flamenco	3683	3069	2029	2302	1842
Cloth-ball	5542	3695	3238	2756	2411

[2] Chitalu F. M., Dubach C., Komura T., "Binary Ostensibly-Implicit Trees for Fast Collision Detection", *Computer Graphics Forum*, vol. 39, no. 2, pp. 509-521, 2020.

[3] Chitalu F. M., Dubach C., Komura T., "Bulk-synchronous Parallel Simultaneous BVH Traversal for Collision Detection on GPUs", *in Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 4-9, 2018.

[4] Curtis S., Tamstorf R., Manocha D., "Fast collision detection for deformable models using representative-triangles", *in Proceedings of the Symposium on Interactive 3D Graphics and Games*, pp. 61-69, 2008.

[5] Du P., Tang M., Tong R. F., "Fast continuous collision culling with deforming noncollinear filters", *Computer Animation and Virtual Worlds*, vol. 23, no. 3, pp. 375-383, 2012.

[6] Du P., Zhao J., et al., "GPU accelerated real-time collision handling in virtual disassembly", *Journal of Computer Science and Technology*, vol. 30, no. 3, pp. 511-518, 2015.

[7] Govindaraju N. K., Knott D., Jain N., et al., "Interactive collision detection between deformable models using chromatic decomposition", *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 991-999, 2005.

[8] Meister D., Bittner J., "Parallel Locally-Ordered Clustering for Bounding Volume Hierarchy Construction", *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 3, pp. 1345-1353, 2018.

[9] Tan T., Weller R.,"OpenCollBench - Benchmarking of Collision Detection & Proximity Queries as a Web-Service", *in Proceedings of the 25th International Conference on 3D Web Technology*, pp. 1-9, 2020.

[10] Tang M., Curtis S., Yoon S. E., "ICCD: interactive continuous collision detection between deformable models using connectivity-based culling", *IEEE Trans on Visualization and Computer Graphics*, vol. 15, no. 4, pp. 544-557, 2009.

[11] Tang M., Manocha D., Tong R. F., "MCCD: Multi-core collision detection between deformable models using front-based decomposition", *Graphical Models*, vol. 72, no. 2, pp. 7-23, 2010.

[12] Tang M., Manocha D., Tong R. F., "Fast continuous collision detection using deforming non-penetration filters", *in Proceedings of the SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pp. 7-13, 2010.

[13] Tang M., Tong R. F., Wang Z. D., et al., "Fast and exact continuous collision detection with bernstein sign classification", *ACM Transactions on Graphics*, vol. 33, no. 6, pp. 1-8, 2014.

[14] Wang H. M., "Defending continuous collision detection against errors", *ACM Transactions on Graphics*, vol. 33, no. 4, pp. 1-10, 2014.

[15] Wang M., Cao J., "A review of collision detection for deformable objects", *Computer Animation and Virtual Worlds*, vol. 32, no. 5, pp.1-23, 2021.

[16] Wang X., et al., "Efficient BVH-based Collision Detection Scheme with Ordering and Restructuring", *Journal of Computer Graphics Forum*, vol. 37, no. 2, pp. 227-237, 2018.

[17] Wong T. H., Leach G., Zambetta F., "An adaptive octree grid for GPU-based collision detection of deformable objects", *Visual Computer*, vol. 30, no. 6, pp. 729-738, 2014.

[18] Wu L., et al., "A Safe and Fast Repulsion Method for GPU-based Cloth Self Collisions", *ACM Transactions on Graphics*, vol. 40, no. 1, pp. 1-18, 2021.

[19] Zhang X., Liu Y., "An algebraic non-penetration filter for continuous collision detection using Sturm theorem", *in Proceedings of the International Conference on Mechatronics and Automation*, pp. 761-766, 2015.

[20] Zhang X., Liu Y., "A fast algebraic non-penetration filter for continuous collision detection", *Graphical Models*, vol. 80, no. 3, pp. 31-40, 2015.

SƠ LƯỢC VỀ TÁC GIẢ



Nghiêm Văn Hưng đang công tác tại Khoa Công nghệ thông tin, Trường Đại học Kỹ thuật – Hậu cần Công an nhân dân. Tốt nghiệp Cử nhân ngành CNTT tại Đại học Sư phạm Đà Nẵng năm 2008, tốt nghiệp Thạc sĩ ngành Hệ thống thông tin tại Học viện Kỹ thuật quân sự năm 2012. Nghiên cứu sinh ngành Hệ thống thông tin tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Lĩnh vực nghiên cứu: Đa phương tiện, Mô phỏng, Thực tại ảo, Hệ thống thông tin.
Điện thoại: 0946.555.189
E-mail: nghiemvanhung1985@gmail.com



Đặng Văn Đức đang công tác tại Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Nhận học vị Tiến sĩ năm 1996, nhận chức danh Phó Giáo sư năm 2002. Lĩnh vực nghiên cứu: Đa phương tiện, GIS, Công nghệ phần mềm.
Điện thoại: 0912.223.163
E-mail: dvduc@ioit.ac.vn



Nguyễn Văn Căn đang công tác tại Trường Đại học Kỹ thuật – Hậu cần Công an nhân dân. Nhận học vị Tiến sĩ chuyên ngành Cơ sở toán học cho tin học năm 2016. Lĩnh vực nghiên cứu: Công nghệ nhận dạng, Thị giác máy tính, Đa phương tiện, An ninh mạng, An toàn thông tin.
Điện thoại: 0986.919.333
E-mail: nguyenvancan@gmail.com



Hoàng Việt Long đang công tác tại Khoa Công nghệ thông tin, Trường Đại học Kỹ thuật – Hậu cần Công an nhân dân. Nhận học vị Tiến sĩ năm 2011, nhận chức danh Phó Giáo sư năm 2018. Lĩnh vực nghiên cứu: Học máy, An ninh mạng, An toàn thông tin, Đa phương tiện.
Điện thoại: 0918.700.525
E-mail: longhv08@gmail.com



Trịnh Hiền Anh đang công tác tại Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Tốt nghiệp Thạc sĩ ngành Công nghệ phần mềm tại Đại học Công nghệ-ĐHQG Hà Nội năm 2011. Nghiên cứu sinh tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Lĩnh vực nghiên cứu quan: Đa phương tiện, Công nghệ mô phỏng, Thực tại ảo, Hệ thống thông tin.
Điện thoại: 0989.197.288
E-mail: hienanh@ioit.ac.vn

Về một phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận topo mờ trực cảm

Trần Thanh Đại[1], Nguyễn Long Giang[2], Trần Thị Ngân[3], Hoàng Thị Minh Châu[4], Vũ Thu Uyên[5], Vương Trung Hiếu[6]
[1,4,5] Trường Đại học Kinh tế Kĩ thuật Công nghiệp, Hà Nội
[2] Viện CNTT - Viện Hàn lâm Khoa học Công nghệ Việt Nam, Hà Nội
[3] Trường Đại học Thủy Lợi, Hà Nội
[6] Trường Đại học Công nghiệp, Hà Nội
Tác giả liên hệ: Trần Thanh Đại, ttdaiuneti@gmail.com
Ngày nhận bài: 14/08/2022, ngày sửa chữa: 15/11/2022, ngày duyệt đăng: 25/11/2022
Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1132

Tóm tắt: Hầu hết các phương pháp rút gọn thuộc tính theo tiếp cận tính toán hạt của tập thô và tập thô mở rộng hiện nay đều sử dụng các độ đo để đánh giá độ quan trọng của thuộc tính cũng như định nghĩa tập rút gọn. Các độ đo này chủ yếu lấy xấp xỉ độ tương tự giữa các hạt thông tin mờ trực cảm mà không thể hiện đầy đủ mức độ tương tự về mặt cấu trúc, do đó tập rút gọn thu được còn chưa hiệu quả về kích thước. Do đó, trong bài báo này chúng tôi đề xuất mô hình rút gọn thuộc tính theo tiếp cận topo mờ trực cảm (Intuitionistic Fuzzy Topology - IFT). Trong đó độ đo độ khác biệt giữa các cơ sở con (subbase) của IFT được định nghĩa để làm công cụ phân loại thuộc tính và cấu trúc cơ sở (base) của IFT đơn vị được sử dụng để định nghĩa tập rút gọn. Các kết quả phân tích về phương diện lý thuyết và thực nghiệm cho thấy phương pháp rút gọn thuộc tính theo tiếp cận IFT cho tập rút gọn có kích thước nhỏ hơn đáng kể so với tiếp cận độ đo truyền thống, trong khi độ chính xác phân lớp của tập rút gọn thu được có thể chấp nhận được trong một số bài toán thực tế.

Từ khóa: *rút gọn thuộc tính, tập thô, tập mờ trực cảm, topo ,topo mờ trực cảm.*

Title: **Intuitionistic Fuzzy Topology-Based Attribute Reduction on the Decision Table**

Abstract: Most attribute reduction methods following the granular computing approach of rough sets and extended rough sets today use metrics to evaluate the attribute's importance and define the reduced set. These measures mainly approximate the similarity between intuitionistic fuzzy granular but do not fully represent the structural similarity so that the resulting reduct could be more efficient in size. This paper proposes an attribute reduction model according to the intuitionistic fuzzy topological approach. The measure of the difference between the subbases is defined to evaluate the attribute's importance, and the unit base of the unit topology is used to define the reduced set. The results of the theoretical and experimental analysis show that the attribute proposed method has a significantly smaller size than the traditional measure approach, while the classification accuracy of the reduct is acceptable in some real problems.

Keywords: *attribute reduction, rough set, intuitionistic fuzzy set, topology, intuitionistic fuzzy topology.*

I. GIỚI THIỆU

Rút gọn thuộc tính là một quá trình tiền xử lý dữ liệu quan trọng trong các lĩnh vực nhận dạng mẫu, khai thác dữ liệu và học máy. Trong thực tế, nhiều thuộc tính (attribute) hay các đặc trưng (features) có thể loại bỏ khỏi tập dữ liệu mà không ảnh hưởng đến chất lượng của mô hình được xây dựng so với tập dữ liệu gốc [2].

Đối với các bảng quyết định có miền giá trị số (liên tục). Các thuật toán hiện nay đều rút gọn trực tiếp trên các bảng quyết định gốc mà không phải trải qua quá trình rời rạc hóa dữ liệu. Khi đó khái niệm các hạt thông tin mờ dựa trên nền tập mờ (Fuzzy Set - FS) [7] và hạt thông tin mờ trực cảm dựa trên nền tập mờ trực cảm (Intuitionistic Fuzzy Set - IFS) [9, 10, 12] được sử dụng. Đây là các khái niệm mở rộng từ hạt thông tin truyền thống để xây dựng các độ đo mở rộng như: miền dương mờ trực cảm (Intuitionistic Fuzzy POS - IFPOS) [10], entropy mờ trực cảm (Intuitionistic Fuzzy Entropy - IFE) [12], khoảng cách mờ trực cảm (Intuitionistic Fuzzy Distance - IFD) [9]. Trên

cơ sở đó, các thuật toán rút gọn thuộc tính trên tập nền FS và IFS được phát triển [9, 10, 12]. Tuy nhiên các thuật toán này vẫn sử dụng các độ đo để đánh giá độ quan trọng của thuộc tính cũng như định nghĩa tập rút gọn. Các độ đo này chủ yếu lấy xấp xỉ độ tương tự giữa các hạt thông tin mờ trực cảm mà không thể hiện đầy đủ mức độ tương tự về mặt cấu trúc, do đó tập rút gọn thu được còn chưa hiệu quả về kích thước.

Gần đây tiếp cận rút gọn thuộc tính trên nền không gian topo được nhiều nhà nghiên cứu quan tâm do những mối liên hệ về cấu trúc giữa tập thô và topo nói chung, tập thô mờ trực cảm (Intuitionistic Fuzzy Rough Set - IFRS) và IFT [2–4] nói riêng. Trong đó, một số phương pháp xây dựng topo đã được đề xuất thông qua các phương pháp tiếp cận như: phương pháp dựa trên quan hệ ưu tiên [11], phương pháp dựa trên tập thô (RS) [2–4]. Tuy nhiên các nghiên cứu này chỉ dừng lại ở khía cạnh lý thuyết, chưa có ứng dụng rút gọn thuộc tính hoàn chỉnh cho bảng quyết định nói chung và đặc biệt đối với bảng quyết định có miền giá trị số/ liên tục nói riêng.

Trong bài báo này chúng tôi sử dụng cấu trúc topo mờ trực cảm để xây dựng mô hình rút gọn thuộc tính theo tiếp cận Filter. Trong đó tập nền IFS được sử dụng do IFS được cộng đồng nghiên cứu đánh giá cao về khả năng xử lý tốt các trường hợp dữ liệu nhiễu [8, 9, 10]. Sử dụng độ đo độ tương tự giữa các cấu trúc cơ sở con mờ trực cảm (IF-subbsae) của IFT để đánh giá thuộc tính và sử dụng cấu trúc cơ sở mờ trực cảm (IF-base) đơn vị của IFT làm điều kiện dừng của thuật toán. Phần còn lại của bài báo được tổ chức như sau:

Phần II nhắc lại một số kiến thức cơ bản về tập nền IFS và cấu trúc IFT. Phần III đề xuất phương pháp xây dựng câu trúc IFT và các phép toán tương ứng trên các cấu trúc IF-base và IF-subbase. Phần IV đề xuất mô hình rút gọn thuộc tính theo tiếp cận Filter. Phần V trình bày một số kết quả thực nghiệm và kết luận được trình bày trong Phần VI của bài báo.

II. CÁC KHÁI NIỆM CƠ BẢN

Trước khi đi vào cấu trúc IFT được đề xuất trong Phần III của bài báo, phần này nhắc lại một số khái niệm về bảng quyết định miền giá trị số, các định nghĩa và các tính chất về tập nền IFS và không gian IFT.

Bảng quyết định miền giá trị số được kí hiệu bởi trong đó O là tập không rỗng các đối tượng, X là tập không rỗng các thuộc tính điều kiện và Y là thuộc tính quyết định. Đối với miền giá trị của mỗi thuộc tính điều kiện trong X là các giá trị số liên tục trong khi miền giá trị của thuộc tính Y là các giá trị rời rạc.

Định nghĩa 2.1: [3] Cho O là một tập khác rỗng các đối tượng. Tập mờ trực cảm (IFS) $A(A_{IF})$ có dạng:

Bảng I
BẢNG QUYẾT ĐỊNH MIỀN GIÁ TRỊ SỐ

U	x_1	x_2	x_3	x_4	x_5	x_6	Y
o_1	1.0	0.4	0.8	0.2	1.0	0.0	0
o_2	1.0	0.4	0.2	0.4	0.2	0.8	1
o_3	0.8	0.6	1.0	0.0	0.6	0.4	0
o_4	0.2	0.6	0.8	0.2	0.0	1.0	1
o_5	0.2	0.8	0.8	0.2	0.0	1.0	1
o_6	0.2	0.8	0.2	0.8	0.0	1.0	0

$$A_{IF} = \left\{\left\langle o, \mu_{A_{IF}}(o), \gamma_{A_{IF}}(o)\right\rangle : o \in O\right\}$$

Với mỗi $o \in O$, độ thành viên của o in A_{IF} được kí hiệu bởi $\mu_{A_{IF}} : O \to [0,1]$, và độ không thành viên của o in A_{IF} được kí hiệu bởi $\gamma_{A_{IF}} : O \to [0,1]$, trong đó $0 \le \mu_{A_{IF}}(o) + \gamma_{A_{IF}}(o) \le 1$.

Mệnh đề 2.1: [3] Cho A_{IF} và B_{IF} là hai tập mờ trực cảm xác định trên O. Ta có:

(1) $A_{IF} \subseteq B_{IF}$ iff $\mu_{A_{IF}}(o) \le \mu_{B_{IF}}(o)$ and $\gamma_{A_{IF}}(o) \ge \gamma_{B_{IF}}(o)$ for all $o \in O$.

(2) $A_{IF} = B_{IF}$ iff $A_{IF} \subseteq B_{IF}$ and $B_{IF} \subseteq A_{IF}$.

(3) $\bar{A}_{IF} = \left\{\left\langle o, \gamma_{A_{IF}}(o), \mu_{A_{IF}}(o)\right\rangle : o \in O\right\}$

(4) $A_{IF} \cap B_{IF} = \left\{\left\langle \begin{array}{l} o, \mu_{A_{IF}}(o) \wedge \mu_{B_{IF}}(o), \\ \gamma_{A_{IF}}(o) \vee \gamma_{B_{IF}}(o) \end{array} \right\rangle : o \in O\right\}$

(5) $A_{IF} \cup B_{IF} = \left\{\left\langle \begin{array}{l} o, \mu_{A_{IF}}(o) \vee \mu_{B_{IF}}(o), \\ \gamma_{A_{IF}}(o) \wedge \gamma_{B_{IF}}(o) \end{array} \right\rangle : o \in O\right\}$

Định nghĩa 2.2: [2] Topo mờ trực cảm (IFT) xác định trên O là một họ $\mathcal{T}$ các tập IFS trên O thỏa mãn các tiên đề sau:

(T_1) $0_{IF}, 1_{IF} \in \mathcal{T}$

(T_2) $G_1 \cap G_2 \in \mathcal{T} : G_1, G_2 \in \mathcal{T}$

(T_3) $\cup\, G_i \in \mathcal{T} : \{G_i : G_i \in \mathcal{T}, i \in I\}$

Khi đó, cặp $(O, \mathcal{T})$ được gọi là không gian IFT.

Định nghĩa 2.3: [2] Cho O là tập không rỗng các đối tượng, khi đó cơ sở (base) của topo $\mathcal{T}$ trên O là một tập hợp $\mathcal{B}$ của O sao cho:

(1) Với mọi $o \in O$, tồn tại $G \in O$ sao cho $o \in G$.

(2) Nếu $o \in G_1 \cap G_2$, thì tồn $G_3 \in B$ sao cho $o \in G_3$. Trong đó $G_3 = G_1 \cap G_2$, và $G_1, G_2 \in B$.

Định nghĩa 2.4: [2] Cho không gian topo mờ trực cảm $(O, \mathcal{T})$. Khi đó $\mathcal{S} \subseteq \mathcal{T}$ được gọi là một cơ sở con (subbase) của $\mathcal{T}$ nếu giao hữu hạn các phần tử của $\mathcal{S}$ tạo thành cơ sở (base) $\mathcal{B}$ của $\mathcal{T}$.

III. CẤU TRÚC TOPO MỜ TRỰC CẢM SỬ DỤNG QUAN HỆ ƯU TIÊN MỜ TRỰC CẢM

Trước khi đi vào đề xuất phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận IFT trong Phần IV của bài báo, phần này đề xuất phương pháp xây dựng các tập cơ sở mờ trực cảm (IF-base) và cơ sở con mờ trực cảm

(IF-subbase). Sau đó là một số phép toán và tính chất hoạt động trên các IF-base và IF-subbase được khảo sát.

Định nghĩa 3.1: Cho bảng quyết định $DT = (O, X, Y)$. Với mọi $(m, n) \in O$ và $\delta \in [0.5, 1]$, quan hệ ưu tiên $IFR^{\geq}$ tương ứng với thuộc tính $a \subseteq X$ được xác định như sau: $IFR_a^{\geq}(m, n) = \langle n, \mu_n, \gamma_n \rangle$

Trong đó:

$$\mu_n = \begin{cases} 1 - |a(m) - a(n)| & if\ p_a(m,n) \geq \delta \\ 0 & if \quad other \end{cases} \qquad (1)$$
$$\gamma_n = 1 - \mu_n$$

Với: $p_a(m, n) = \frac{a(m) - a(n) + 1}{2}$; $a(m)$ và $a(n)$ là các giá trị của m và n tương ứng với thuộc tính a.

Định nghĩa 3.2: Cho bảng quyết định $DT = (O, X, Y)$ và quan hệ ưu tiên mờ trực cảm $IFR_a^{\geq}$ tương ứng với thuộc tính $a \subseteq X$. Khi đó với mọi $(x, y) \in O$, quan hệ $IFR_a^{\geq}$ có thể được biểu diễn dưới ma trận quan hệ mờ trực cảm như sau: $\mathcal{M}_a = \left[IFR_a^{\geq}(x, y)\right]_{O \times O}$.

Ví dụ 1: Cho bảng quyết định $DT = (O, X, Y)$ như trong Bảng I, với $\delta = 0.5$ ta có:

$$M_{x_1}^{\geq} = \begin{bmatrix} (1,0) & (1,0) & (0.8,0.2) & (0.2,0.8) & (0.2,0.8) & (0.2,0.8) \\ (1,0) & (1,0) & (0.8,0.2) & (0.2,0.8) & (0.2,0.8) & (0.2,0.8) \\ (0,1) & (0,1) & (1,0) & (0.4,0.6) & (0.4,0.6) & (0.4,0.6) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \end{bmatrix}$$

Định nghĩa 3.3: Cho bảng quyết định $DT = (O, X, Y)$ và ma trận quan hệ $\mathcal{M}_a^{\geq}$, khi đó IF-subbase S_a được xác định như sau: $\mathcal{S}_a = \{\mathcal{S}_a^L, S_a^R\}$.

Trong đó $\mathcal{S}_a^L = \mathcal{M}_a^{\geq}$ và $\mathcal{S}_a^R = \left(\mathcal{M}_a^{\geq}\right)^T$, với $\left(\mathcal{M}_a^{\geq}\right)^T$ là ma trận chuyển vị của ma trận $\mathcal{M}_a^{\geq}$.

Ví dụ 2: Tiếp theo Ví dụ 1, ta có:

$$\left(\mathcal{M}_{x_1}^{\geq}\right)^T = \begin{bmatrix} (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (0.8,0.2) & (0.8,0.2) & (1,0) & (0,1) & (0,1) & (0,1) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \\ (0.2,0.8) & (0.2,0.8) & (0.4,0.6) & (1,0) & (1,0) & (1,0) \end{bmatrix}$$

Định nghĩa 3.4: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_p, \mathcal{S}_q$ tương ứng của $p, q \in X$. Khi đó giao của $\mathcal{S}_p$ và $\mathcal{S}_q$ được xác định như sau:

$$\mathcal{S}_p \cap \mathcal{S}_q = \{\mathcal{S}_p^L \cap \mathcal{S}_q^L, \mathcal{S}_p^R \cap \mathcal{S}_q^R\} \qquad (2)$$

Định nghĩa 3.5: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_p, \mathcal{S}_q$ tương ứng của $p, q \in X$. Khi đó hợp của $\mathcal{S}_p$ và $\mathcal{S}_q$ được xác định như sau:

$$\mathcal{S}_p \cup \mathcal{S}_q = \{\mathcal{S}_p^L \cup \mathcal{S}_q^L, \mathcal{S}_p^R \cup \mathcal{S}_q^R\} \qquad (3)$$

Định nghĩa 3.6: Cho bảng quyết định $DT = (O, X, Y)$ và $\mathcal{S}_a = \{\mathcal{S}_a^L, S_a^R\}$, trong đó $\mathcal{S}_a^L$ là IF-subbase trái, và $\mathcal{S}_a^R$ là IF-subbase phải tương ứng của $a \in X$. Khi đó IF-base $\mathcal{B}_a$ được xác định như sau:

$$\mathcal{B}_a = \mathcal{S}_a{}^L \cap \mathcal{S}_a{}^R \qquad (4)$$

Mệnh đề 3.1: Cho bảng quyết định $DT = (O, X, Y)$ và IF-subbase $\mathcal{B}_a$ tương ứng với thuộc tính $a \in X$ được định nghĩa bởi công thức (4). Khi đó $\mathcal{B}_a$ là một cơ sở của IFT $\mathcal{T}_a$.

Ví dụ 3: Tiếp theo ví dụ 2, ta có:

$$\mathcal{B}_{x_1} = \begin{bmatrix} (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (1,0) & (1,0) & (0,1) & (0,1) & (0,1) & (0,1) \\ (0,1) & (0,1) & (1,0) & (0,1) & (0,1) & (0,1) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \\ (0,1) & (0,1) & (0,1) & (1,0) & (1,0) & (1,0) \end{bmatrix}$$

IV. PHƯƠNG PHÁP RÚT GỌN THUỘC TÍNH THEO TIẾP CẬN TOPO MỜ TRỰC CẢM

Trong phần này chúng tôi đề xuất thuật toán rút gọn thuộc tính theo tiếp cận Filter, sử dụng cấu trúc IFT được định nghĩa trong Phần III của bài báo. Trong đó độ đo độ khác biệt giữa các IF-subbase được xây dựng để đánh giá thuộc tính và sử dụng IF-base của IFT đơn vị để làm điều kiện dừng của thuật toán. Cuối cùng là phần đánh giá độ phức tạp của thuật toán.

1. Độ đo độ khác biệt giữa các cơ sở con của topo mờ trực cảm

Định nghĩa 4.1: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_p, \mathcal{S}_q$ tương ứng của thuộc tính $p, q \in X$. Khi độ khác biệt của $\mathcal{S}_p$ và $\mathcal{S}_q$ được xác định như sau:

$$\begin{aligned} &\zeta(\mathcal{S}_p, \mathcal{S}_q) \\ &= \frac{1}{|O|} \sum_{i=1}^{|O|} \left(\left|\mathcal{S}_p^L[i] \cup S_q^L[i]\right| - \left|\mathcal{S}_p^L[i] \cap S_q^L[i]\right| \right) \\ &+ \frac{1}{|O|} \sum_{i=1}^{|O|} \left(\left|\mathcal{S}_p^R[i] \cup S_q^R[i]\right| - \left|\mathcal{S}_p^R[i] \cap S_q^R[i]\right| \right) \end{aligned} \qquad (5)$$

Mệnh đề 4.1: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_p, \mathcal{S}_q$ tương ứng của thuộc tính $p, q \in X$. Khi độ công thức (5) là tương dương với công thức (6) sau đây:

$$\begin{aligned} &\zeta(\mathcal{S}_p, \mathcal{S}_q) \\ &= \frac{2}{|O|} \sum_{i=1}^{|O|} \left(\left|\mathcal{S}_p^L[i] \cup \mathcal{S}_q^L[i]\right| - \left|\mathcal{S}_p^L[i] \cap \mathcal{S}_q^L[i]\right| \right) \end{aligned} \qquad (6)$$

Mệnh đề 4.2: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_X, \mathcal{S}_{X \cup Y}$. Khi đó:

$$\zeta(\mathcal{S}_X, \mathcal{S}_{X \cup Y}) = \frac{2}{|O|} \sum_{i=1}^{|O|} \left(\left|\mathcal{S}_Y^L[i] - \mathcal{S}_Y^L[i] \cap \mathcal{S}_X^L[i]\right| \right) \qquad (7)$$

là độ đo khác biệt giữa $\mathcal{S}_X$ và $\mathcal{S}_{X \cup Y}$.

Mệnh đề 4.3: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_X, \mathcal{S}_R$, với $R \subseteq X$. Khi đó:

$$\zeta(\mathcal{S}_Y, \mathcal{S}_{Y \cup X}) \leq \zeta(\mathcal{S}_Y, \mathcal{S}_{Y \cup R})$$

2. Rút gọn thuộc tính cho bảng quyết định sử dụng cấu trúc topo mờ trực cảm.

Định nghĩa 4.2: Cho bảng quyết định $DT = (O, X, Y)$ và $R \subseteq X$. Khi đó độ quan trọng của thuộc tính $a \in X$ với R được xác định như sau:

$$Sig_R(a) = \zeta(\mathcal{S}_Y, \mathcal{S}_{Y \cup R \cup a}) - \zeta(\mathcal{S}_Y, \mathcal{S}_{Y \cup R}) \quad (8)$$

Mệnh đề 4.4: Cho bảng quyết định $DT = (O, X, Y)$ và hai IF-subbase $\mathcal{S}_X, \mathcal{S}_R$, với $R \subseteq X$. Khi đó: nếu $\mathcal{B}_R = \mathcal{B}_I$ thì $\mathcal{B}_X = \mathcal{B}_I$.

Trong đó B_I là là một IF-base của IFT đơn vị, với B_I được xác định như sau:

$$\underset{0 \le i,j \le |O|}{B_I}[i,j] = \begin{cases} (1,0) : i = j \\ (0,1) : i \neq j \end{cases}$$

Định nghĩa 4.3: Cho bảng quyết định $DT = (O, X, Y)$ và $R \subseteq X$. Khi đó R được gọi là tập rút gọn của X khi và chỉ khi:

(1) $\mathcal{B}_R = \mathcal{B}_I$;

(2) $\mathcal{B}_{R-c} \neq \mathcal{B}_I$ với mọi $c \in R$.

Thuật toán 1: Thuật toán Filter thuộc tính theo tiếp cận topo mờ trực cảm (F_IFT).

```
1  Dữ liệu vào: Decision Table DT = (O, X, Y) với δ = 0.5.
2  Dữ liệu ra: Tập rút gọn R.
3  begin
4      R ← ∅;
5      B_R ← [1_~]_{O×O};
6      Init B_I;
7      for c ∈ X ∪ Y do
8          calculate S_c dựa trên công thức (1);
9      end
10     while B_R ≠ B_I do
11         for c ∈ X − R do
12             calculate Sig_R(c) dựa trên công thức (8);
13         end
14         select c_m ∈ X − R : c_m = Max_{c∈X−R} {Sig_R(c)};
15         R ← R ∪ {c_m};
16         update B_R dựa trên công thức (4);
17     end
18     Return R;
19 end
```

Tiếp theo chúng tôi đánh giá độ phức tạp tính toán của giải thuật rút gọn đề xuất F_IFT. Trước tiên chúng tôi kí hiệu $|X|, |O|$ tương ứng là kích thước của tập thuộc tính X và kích thước của tập đối tượng O:

(1). Độ phức tạp tính toán tại các dòng lệnh 7-9 là $O\left(|X|\,|O|^2\right)$;

(2). Độ phức tạp tính toán tại các dòng lệnh 11-13 là $O\left(|X-R|\,|O|^2\right)$, độ phức tạp tại dòng lệnh 14-15 $O(|X-R|)$, độ phức tạp tại dòng lệnh 16 là $O\left(|O|^2\right)$. Do đó độ phức tạp tính toán của vòng lặp **while** là: $O\left(|X-R|\,|O|^2\right)$;

Từ (1) and (2), ta có độ phức tạp tính toán của thuật toán F_IFT là $O\left(|X|\,|O|^2\right)$.

V. MỘT SỐ KẾT QUẢ THỰC NGHIỆM

Để củng cố cơ sở lý thuyết về phương pháp rút gọn thuộc tính cho bảng quyết định theo tiếp cận IFT và đánh giá hiệu năng của thuật toán đề xuất F_IFT được trình bày trong Phần IV của bài báo, phần này trình bày kế hoạch thực nghiệm và các kết quả thực nghiệm của thuật toán F_IFT trên các bộ dữ liệu thực.

Bảng II
BẢNG MÔ TẢ DỮ LIỆU THỰC NGHIỆM

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

1. Kế hoạch và môi trường thực nghiệm

Việc thực nghiệm nhằm đánh giá tính hiệu quả của thuật toán đề xuất bằng cách so sánh thuật toán đề xuất F_IFT với các thuật toán F_IFPOS [10], F_IFE [12], và F_IFD [9]. Tất cả các thuật toán trên đều được xây dựng theo tiếp cận Filter thuộc tính và sử dụng tập nền IFS để xây dựng các độ đo. Môi trường thực nghiệm cho các thuật toán được thực hiện trên nền hệ điều hành Window 10, CPU core i5 Processor – RAM 8Gb. Tất cả các thuật toán đều được cài đặt bằng ngôn ngữ lập trình Python và thực nghiệm với 12 bộ dữ liệu thực từ UCI [8]. Chi tiết về các tập dữ liệu được mô tả trong Bảng II.

Để đánh giá các thuật toán, chúng tôi sử dụng ba tiêu chí để đánh giá đó là:

1) Kích thước của tập rút gọn;

2) Độ chính xác phân lớp thuộc đoạn [0,1];

3) Thời gian thực hiện của thuật toán (s).

Trong đó độ chính xác phân lớp được thống kê trên hai mô hình phân lớp SVM và K-láng giềng gần (KNN)

với số lượng láng giềng là K=10 và $p = 2$ là khoảng cách Euclidean được sử dụng. Độ đo *accuracy* kết hợp với phương pháp đánh giá chéo 10-fold được sử dụng trên từng mô hình phân lớp. Các tập dữ liệu trước khi rút gọn được chuẩn hóa về miền giá trị [0,1] và công thức (1) được sử dụng với $\delta = 0.5$. Kết quả so sánh trong các Bảng là giá trị trung bình của 10 lần chạy với 10 lần lấy mẫu ngẫu trên trên từng tập dữ liệu.

2. Đánh giá thuật toán F_IFT

Bảng III mô tả các thuộc tính của tập rút gọn thu được từ thuật toán đề xuất F_IFT. Bảng IV so sánh kích thước của tập rút gọn gữa thuật toán đề xuất F_IFT so với nhóm các thuật toán Filter được mô tả ở bên trên. Trong khi đó Bảng V và Bảng VI được dùng để so sánh độ chính xác phân lớp trên hai mô hình SVM và KNN giữa các tập rút gọn thu được. Bảng VII được dùng để so sánh về thời gian tính toán tập rút gọn giữa các thuật toán.

Như chúng ta có thể quan sát thấy tại Bảng IV, thuật toán đề xuất F_IFT cho tập rút gọn có kích thước tốt hơn đáng kể so với các thuật toán khác. Thông qua kích thước trung bình (5.33) trên 12 tập dữ liệu, chúng ta có thể khẳng định thuật toán đề xuất F_IFT cho tập rút gọn có kích thước tốt hơn so với các thuật toán trên toàn bộ 12 tập dữ liệu thực nghiệm. Hơn nữa thời gian thực hiện của thuật toán đề xuất F_IFT cũng rất tốt, cụ tể với 12 tập dữ liệu thực nghiệm, thuật toán đề xuất F_IFT có thời gian thực hiện trung bình nhỏ nhất (1.25s). Điều này có thể lý giải được là do tốc độ hội tụ của thuật toán tỉ lệ thuận với kích thước của tập rút gọn thu được.

Tuy nhiên, độ chính xác phân lớp của tập rút gọn thu được từ thuật toán đề xuất F_IFT còn thấp hơn so với các thuật toán khác trên cả hai mô hình SVM và KNN được xử dụng, điều này có thể lý giải do số lượng thuộc tính của tập rút gọn khá nhỏ, ảnh hưởng tới chất lượng phân lớp của mô hình. Thông qua độ chính xác trung bình (0.7) với SVM và (0.71) với KNN, chúng ta có thể khẳng định việc đánh đổi giữa kích thước của tập rút gọn và độ chính xác phân lớp hoàn toàn có thể chấp nhận được vì độ chính xác phân lớp của mô hình vẫn có thể chấp nhận được trong một số bài toán trong thực tế.

Bảng III
TẬP RÚT GỌN THU ĐƯỢC TỪ THUẬT TOÁN F_IFT

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

Bảng IV
BẢNG SO SÁNH KÍCH THƯỚC GIỮA CÁC TẬP RÚT GỌN

ID	Data	F_IFT	[9]	[10]	[12]	$\|X\|$
1	Wine	3	10	13	11	13
2	Heart	3	13	13	13	13
3	Wdbc	2	16	21	7	30
4	Wpbc	2	14	22	18	33
5	Iono	16	16	29	17	34
6	UFDC	6	6	16	5	43
7	Sona	2	10	44	12	60
8	Libras	9	10	22	30	90
9	Musk	8	8	61	10	166
10	LVB	2	5	60	5	310
11	LVG	6	6	60	6	310
12	PD	5	23	90	10	754
Average		5.33	11.42	37.58	12.00	100.18

Bảng V
BẢNG SO SÁNH ĐỘ CHÍNH XÁC PHÂN LỚP TRÊN MÔ HÌNH SVM CỦA CÁC THUẬT TOÁN.

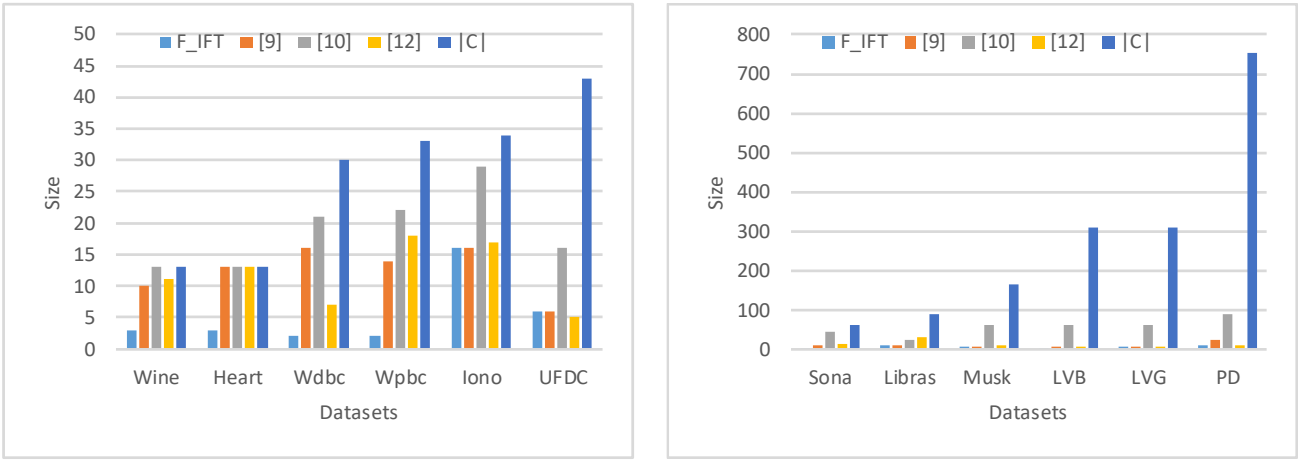
ID	Data	F_IFT	[9]	[10]	[12]	$\|X\|$
1	Wine	0.89	0.97	0.98	0.93	0.98
2	Heart	0.75	0.84	0.84	0.84	0.84
3	Wdbc	0.8	0.96	0.97	0.92	0.98
4	Wpbc	0.77	0.76	0.77	0.76	0.78
5	Iono	0.87	0.84	0.88	0.86	0.88
6	UFDC	0.36	0.35	0.55	0.43	0.43
7	Sona	0.64	0.64	0.67	0.72	0.65
8	Libras	0.57	0.59	0.62	0.65	0.71
9	Musk	0.66	0.63	0.72	0.72	0.75
10	LVB	0.67	0.67	0.86	0.68	0.83
11	LVG	0.67	0.67	0.85	0.82	0.89
12	PD	0.84	0.84	0.88	0.86	0.81
Average		0.70	0.72	0.79	0.76	0.79

VI. KẾT LUẬN

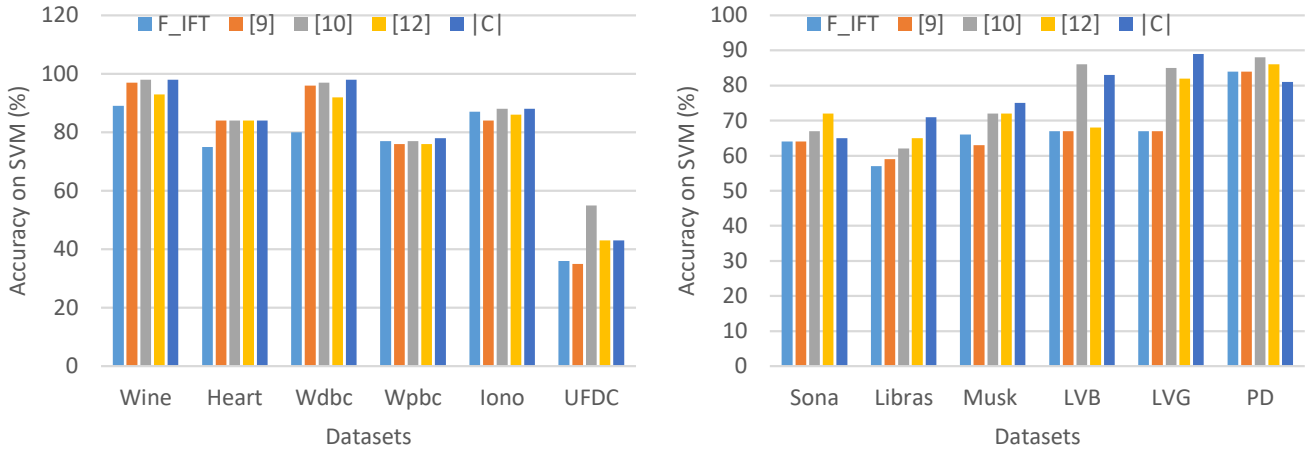
Trong bài báo này, chúng tôi đã sử dụng cấu trúc IFT để rút gọn thuộc tính cho bảng quyết định miền gá trị số. Trong đó, cấu trúc IF-subbase được sử dụng để xây dựng độ đo đánh giá thuộc tính và cấu trúc IF-base của IFT đơn vị được sử dụng để làm điều kiện dừng cho thuật toán. Kết quả thực nghiệm cho thấy phương pháp đề xuất là hiệu quả về thời gian tính toán và kích thước của tập rút gọn. Điều đó càng khẳng định phương pháp chọn điều kiện dừng cho thuật toán đề xuất F_IFT theo tiếp cận IFT là hoàn toàn phù hợp với cách định nghĩa tập rút gọn được phân tích trong Phần IV của bài báo.

Tuy nhiên độ chính xác phân lớp còn hạn chế so với các thuật toán khác, do đó chúng ta có thể cân nhắc việc đánh đổi độ chính xác phân lớp để có tập rút gọn có kích thước tốt và thời gian tính toán hiệu quả.

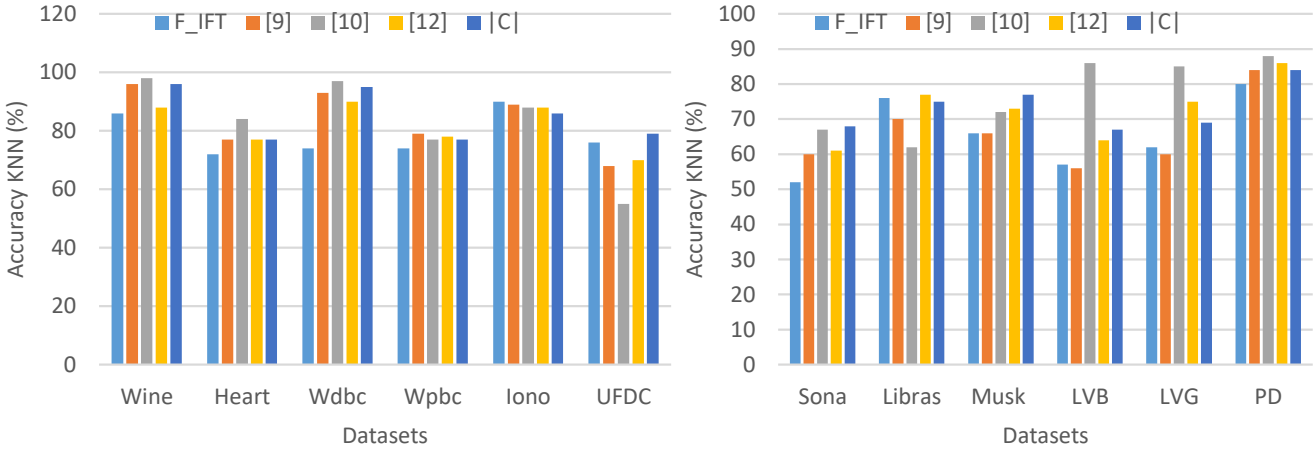
Hơn nữa, cách tiếp cận chọn lọc thuộc tính cho tập rút



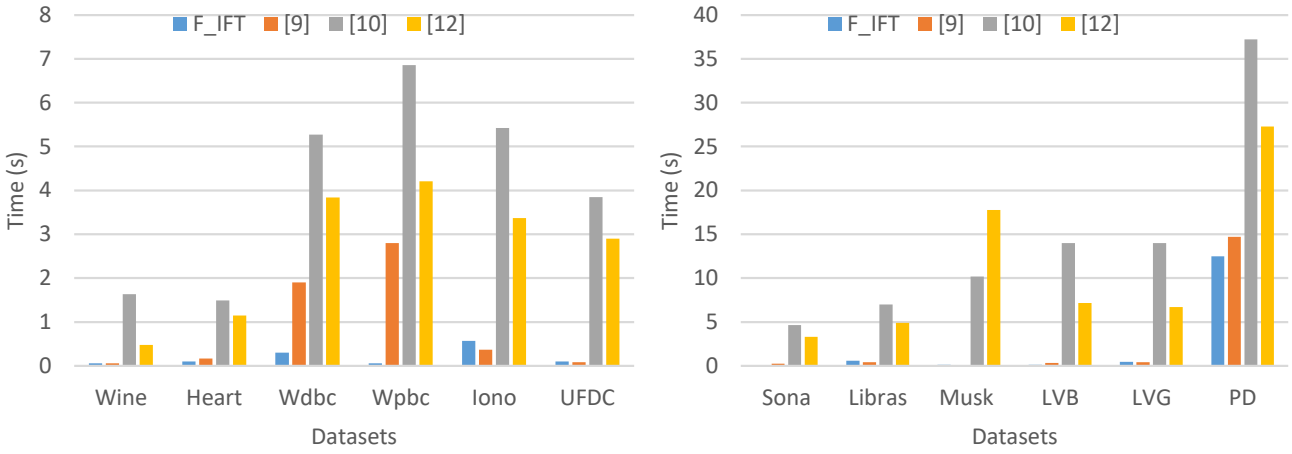
Hình 1. So sánh kích thước của các tập rút gọn thu được



Hình 2. So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình SVM



Hình 3. So sánh độ chính xác phân lớp của các tập rút gọn trên mô hình KNN



Hình 4. So sánh thời gian tính toán tập rút gọn của các thuật toán

Bảng VI
BẢNG SO SÁNH ĐỘ CHÍNH XÁC PHÂN LỚP TRÊN MÔ HÌNH KNN CỦA CÁC THUẬT TOÁN.

ID	Data	F_IFT	[9]	[10]	[12]	$\|X\|$
1	Wine	0.86	0.96	0.98	0.88	0.96
2	Heart	0.72	0.77	0.84	0.77	0.77
3	Wdbc	0.74	0.93	0.97	0.9	0.95
4	Wpbc	0.74	0.79	0.77	0.78	0.77
5	Iono	0.9	0.89	0.88	0.88	0.86
6	UFDC	0.76	0.68	0.55	0.7	0.79
7	Sona	0.52	0.6	0.67	0.61	0.68
8	Libras	0.76	0.7	0.62	0.77	0.75
9	Musk	0.66	0.66	0.72	0.73	0.77
10	LVB	0.57	0.56	0.86	0.64	0.67
11	LVG	0.62	0.6	0.85	0.75	0.69
12	PD	0.82	0.84	0.88	0.86	0.84
Average		0.71	0.74	0.79	0.76	0.79

Bảng VII
BẢNG SO SÁNH THỜI GIAN TÍNH TOÁN TẬP RÚT GỌN GIỮA CÁC THUẬT TOÁN.

ID	Data	F_IFT	[9]	[10]	[12]
1	Wine	0.06	0.06	1.63	0.48
2	Heart	0.1	0.17	1.49	1.15
3	Wdbc	0.3	1.9	5.27	3.84
4	Wpbc	0.06	2.8	6.86	4.21
5	Iono	0.57	0.37	5.42	3.37
6	UFDC	0.1	0.08	3.85	2.9
7	Sona	0.07	0.24	4.63	3.31
8	Libras	0.57	0.43	7	4.9
9	Musk	0.12	0.07	10.2	17.77
10	LVB	0.12	0.34	14	7.15
11	LVG	0.45	0.4	14	6.68
12	PD	12.5	14.7	37.2	27.3
Average		1.25	1.80	9.30	6.92

gọn trong bài vẫn dừng lại ở tiếp cận độ đo, chỉ có điều kiện dừng của thuật toán là sử dụng tiếp cận IFT. Do đó, trong tương tai chúng tôi sẽ xây dựng phương pháp chọn lọc thuộc tính theo tiếp cận IFT và phát triển mô hình lai ghép filter-wrapper để tập rút gọn thu được tốt cả về kích thước và độ chính xác phân lớp.

TÀI LIỆU THAM KHẢO

[1] Atanassov, K. T. (1986). Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*. An International Journal in Information Science and Engineering, 20(1), 87–96.

[2] Yun, S. M., Eom, Y. S., & Lee, S. J. (2021). Topology of the redefined intuitionistic fuzzy rough sets. *International Journal of Fuzzy Logic and Intelligent Systems*, 21(4), 369–377.

[3] Yun, S. M., Eom, Y. S., & Lee, S. J. (2021). Topology of the redefined intuitionistic fuzzy rough sets. *International Journal of Fuzzy Logic and Intelligent Systems*, 21(4), 369–377.

[4] Bashir, Z., Abbas Malik, M. G., Asif, S., & Rashid, T. (2020). The topological properties of intuitionistic fuzzy rough sets. *Journal of Intelligent & Fuzzy Systems*, 38(1), 795–807.

[5] Abo-Elhamayel, M., & Yang, Y. (2021). Generalizations of rough sets via topology. *Afrika Matematika*, 32(1–2), 41–50.

[6] Rebecca Paul, N. (2019). Rough sets applicable in a topology. *Malaya Journal of Matematik*, S(1), 61–65.

[7] Giang, N. L., Son, L. H., Ngan, T. T., Tuan, T. M., Phuong, H. T., Abdel-Basset, M., de Macedo, A. R. L., & de Albuquerque, V. H. C. (2020). Novel incremental algorithms for attribute reduction from dynamic decision tables using hybrid filter–wrapper with fuzzy partition distance. *IEEE Transactions on Fuzzy Systems*: A Publication of the IEEE Neural Networks Council, 28(5), 858–873.

[8] UCI. machine learning repository (2021). Data. https://archive.ics.uci.edu/ml/index.php

[9] Nguyen, T. T., Giang, N. L., Tran, D. T., Nguyen, T. T., Nguyen, H. Q., Pham, A. V., & Vu, T. D. (2021). A novel filter-wrapper algorithm on intuitionistic fuzzy set for attribute reduction from decision tables. *International journal of data warehousing and mining*, 17(4), 67–100.

[10] Tan, A., Wu, W.-Z., Qian, Y., Liang, J., Chen, J., & Li, J. (2019). Intuitionistic fuzzy rough set-based granular structures and attribute subset selection. *IEEE Transactions on Fuzzy Systems*: A Publication of the IEEE Neural Networks Council, 27(3), 527–539.

[11] Mishra, S., & Srivastava, R. (2018). Fuzzy topologies are generated by fuzzy relations. *Soft Computing*, 22(2), 373–385.

[12] Tan, A., Shi, S., Wu, W.-Z., Li, J., & Pedrycz, W. (2022). Granularity and entropy of intuitionistic fuzzy information and their applications. *IEEE Transactions on Cybernetics*, 52(1), 192–204.

SƠ LƯỢC VỀ TÁC GIẢ



Trần Thanh Đại nhận bằng Thạc sỹ khoa học máy tính năm 2011 tại Học viện Kĩ thuật Quân Sự. Hiện là nghiên cứu sinh tại Đại học Khoa học Công nghệ - Viện Hàn Lâm Khoa học Công nghệ Việt Nam.
Lĩnh vực nghiên cứu: Tính toán mềm, tính toán hạt, tính toán thô, khai phá dữ liệu và học máy.
Email: ttdaiuneti@gmail.com



Nguyễn Long Giang nhận bằng Tiến Sỹ Toán học năm 2012, Phó Giáo sư tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học Việt Nam năm 2017. Tổng biên tập tạp chí Tin học và điều khiển (JCC). Tham gia điều hành tại các hội thảo quốc tế như: IJCRS 2019, AICI 2019, MARR 2019, IEEE-RIVF 2019, SoICT 2019.
Lĩnh vực nghiên cứu: Trí tuệ nhân tạo (AI), tính toán mềm, tính toán mờ, khai phá dữ liệu:
Email: nlgiang@ioit.ac.vn



Trần Thị Ngân nhận bằng Đại học toán ứng dụng tại Trường Đại học Khoa học tự nhiên - Đại học Quốc Gia năm 2003, bằng Thạc sỹ khoa học máy tính năm 2007 tại Đại học Thái Nguyên và bằng Tiến sỹ năm 2014. Được phong Phó giáo sư năm 2021. Hiện đang công tác tại Khoa công nghệ thông tin - Trường Đại học Thủy Lợi.
Lĩnh vực nghiên cứu: Tính toán mờ, tối ưu, khai phá dữ liệu và học máy.
Email: ngantt@tlu.edu.vn



Hoàng Thị Minh Châu nhận bằng Thạc sỹ Bảo đảm toán học cho máy tính và hệ thống tính toán năm 2010 tại Đại học Khoa Học Tự Nhiên – ĐHQGHN. Hiện đang là giảng viên khoa Công nghệ thông tin - Đại học Kinh tế Kỹ thuật Công nghiệp.
Lĩnh vực nghiên cứu: IOT, AI, Khai phá dữ liệu và học máy.
Email: htmchau@uneti.edu.vn



Vũ Thu Uyên nhận bằng Thạc sỹ nghành công nghệ phần mềm năm 2009 tại Trường Đại học công nghệ - Đại học Quốc Gia. Hiện là giảng viên Khoa Công nghệ thông tin - Trường Đại học Kinh tế Kĩ thuật Công nghiệp.
Lĩnh vực nghiên cứu chính: Khai phá dữ liệu và học máy
Email: vtuyen@uneti.edu.vn



Vương Trung Hiếu nhận bằng Cử nhân ngành Kỹ thuật máy tính tại Đại học Osmania, Ấn Độ. Nghiên cứu sinh tại Học viện Khoa học Công nghệ - Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Hiện đang là viên chức Phòng Tổ chức - Hành chính, Trường Đại học Công nghiệp Hà Nội.
Lĩnh vực nghiên cứu: Tính toán mềm, khai phá dữ liệu và học máy.
Email: vthieu@haui.edu.vn

Một kỹ thuật định vị trong nhà bằng WiFi hiệu quả sử dụng học máy kết hợp

Ngô Văn Bình[1], Vũ Văn Hiệu[2]
[1] Trường Đại học FPT, Hà Nội
[2] Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam
Tác giả liên hệ: Vũ Văn Hiệu, vvhieu@ioit.ac.vn
Ngày nhận bài: 21/08/2022, ngày sửa chữa: 15/11/2022, ngày duyệt đăng: 25/11/2022
Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1124

Tóm tắt: Trong nghiên cứu này, chúng tôi đề xuất mô hình học tích hợp nhiều mô hình học máy theo hai pha. Pha thứ nhất kết hợp tổ hợp các mô hình phân lớp k-Nearest Neighbor (KNN), Logistic Regression (LR) , Support Vector Machines (SVM) được tổ hợp dữ liệu mới. Pha thứ hai sử dụng một bộ phân lớp trên tổ hợp dữ liệu mới. Mô hình này được áp dụng trong giai đoạn huấn luyện mô hình. Trong giai đoạn kiểm thử, dữ liệu thu từ thiết bị điện thoại được tiền xử lý, sau đó dữ liệu được đưa vào mô hình dự báo để ước tính một vị trí tòa-tầng cụ thể trong tòa nhà. Đề xuất được cài đặt và đánh giá trên tập dữ liệu UJIIndoorLoc đạt kết quả dự báo chính xác 98.73%. Kết quả này cao hơn kết quả của các mô hình sử dụng từng thuật toán thành phần và cũng cao hơn kết quả của các nghiên cứu khác dùng cùng thuật toán và trên cùng tập dữ liệu.

Từ khóa: *dấu vết WiFi, cường độ tín hiệu thu-RSS, hệ thống định vị trong nhà, học máy.*

Title: **An Efficient WiFi Indoor Positioning Technique using Combined Machine Learning**

Abstract: In this study, we propose a learning model that integrates many machine learning models in two phases. The first phase combines the combination of k-Nearest Neighbor (KNN), Logistic Regression (LR) and Support Vector Machines (SVM) classification models to combine new data. The second phase uses a classifier on the new dataset. This model is applied during the model training phase. During the testing phase, the data collected from the mobile device is preprocessed, then the data is fed into a predictive model to estimate a specific floor location in the building. The proposal is installed and evaluated on the UJIIndoorLoc dataset with accurate prediction results of 98.73%. This result is higher than the results of models using each component algorithm and also higher than the results of other studies using the same algorithm and on the same dataset.

Keywords: *WiFi fingerPrinting, Received Signal Strength-RSS, indoor positioning system, machine learning.*

I. ĐẶT VẤN ĐỀ

Hệ thống định vị trong nhà (IPS) rất hữu ích trong thực tế. Trong các tòa nhà, IPS ngoài việc trợ giúp điều hướng trong cứu hộ, khẩn cấp nó còn hữu ích trong các môi trường như bảo tàng, bệnh viện, mega mall. Do đó, IPS đã trở thành một lĩnh vực nghiên cứu rộng rãi trong hơn 20 năm qua.

Các phương pháp định vị trong nhà được giới thiệu trong tài liệu [1]. Trong đó, phương pháp dựa trên cường độ tín hiệu (Received Signal Strength - RSS) của sóng radio được phát ra từ các điểm truy cập (Access point - AP) và thu được từ các điểm tham chiếu (Reference Point-RP) được sử dụng nhiều nhất và phổ biến trong các hệ thống định vị trong nhà. Có rất nhiều kỹ thuật định vị trong nhà dựa trên giá trị RSS, trong số đó fingerPrinting là kỹ thuật được dùng hết sức phổ biến [2, 3].

Có hai giai đoạn trong fingerprinting: giai đoạn huấn luyện và giai đoạn kiểm thử. Giai đoạn huấn luyện: thực hiện đo các chỉ số RSS của các AP từ các RP đã biết để xây dựng cơ sở dữ liệu fingerprinting. Giai đoạn kiểm thử, giá trị RSS được lấy mẫu theo thời gian thực tại vị trí chưa biết, giá trị này được so sánh với cơ sở dữ liệu fingerprinting để tìm ra vị trí định vị phù hợp nhất.

IPS dùng fingerPrinting đối mặt với hai vấn đề chính. Thứ nhất, các vector RSS thu được có giá trị biến đổi bất thường, không ổn định dẫn đến số lượng dữ liệu lớn nhưng không chất lượng [4]. Để khắc phục vấn đề này một số phương pháp lọc và giảm kích thước dữ liệu như phương pháp phân tích thành phần chính (Principal

Component Analysis-PCA) [5] và phân tích biệt thức tuyến tính (Linear Discriminant Analysis-LDA) [6] đã được áp dụng nhằm tăng chất lượng tập dữ liệu. Thách thức thứ hai là chất lượng định vị cần có độ chính xác và hiệu quả cao. Nhiều thuật toán học máy như KNN, SVM, LR, Decision Tree(DT), Random Forest (RF), Light Gradient Boosting Machine (LGBM), Convolutions Neural Network (CNN), Deep Neural Networks (DNN), Recurrent Neural Network (RNN), ... đã được thử nghiệm nhằm nâng cao chất lượng định vị và đã đạt được nhiều kết quả đáng kể, tuy nhiên cho đến hiện tại chưa có thuật toán nào được cho là có thể áp dụng cho nhiều môi trường [7, 8].

Trong nghiên cứu này chúng tôi đề xuất phương pháp giảm kích thước dữ liệu và phương pháp định vị bằng mô hình mới, trong đó chúng tôi tích hợp các thuật toán học máy KNN [9, 10], SVM [11], LR [12] để dự báo tòa-tầng bằng bài toán phân lớp, các mô hình được thực nghiệm trên tập dữ liệu UJIIndoorLoc[1] [13]. Cụ thể những đóng góp của bài báo như sau:

- Tiền xử lý dữ liệu trong giai đoạn kiểm thử
- Đề xuất phương pháp giảm kích thước dữ liệu cho bài toán phân lớp ước lượng vị trí tòa nhà và tầng
- Đề xuất mô hình kết hợp các thuật toán học máy phân lớp áp dụng vào bài toán ước lượng vị trí tầng bằng tổ hợp các thuật toán phân lớp LR, KNN, SVM

Bài báo có cấu trúc như sau: Phần I đặt vấn đề, Phần II về các nghiên cứu liên quan, Phần III mô hình đề xuất. Phần IV thực nghiệm và kết quả mới, Phần V kết luận.

II. CÁC NGHIÊN CỨU LIÊN QUAN

Các phương pháp học máy truyền thống được áp dụng trong bài toán định vị trong nhà dùng RSS của sóng WiFi được sử dụng rất nhiều. Có thể coi KNN là thuật toán học máy được sử dụng đầu tiên và mở ra thời kỳ áp dụng học máy vào bài toán định vị trong nhà. Năm 2000 nhóm nghiên cứu của Microsoft Research đã sử dụng KNN để phát triển hệ thống định vị có tên RADAR [9], kết quả KNN tốt hơn rất nhiều thuật toán fingerprinting. Trong [14] các tác giả đã sử dụng KNN kết hợp với thông tin vị trí người dùng trong quá khứ. Kết quả cho thấy phương pháp mới đạt hiệu suất định vị cao hơn 45% so với KNN. Các tác giả của nghiên cứu [15] kết hợp DNN và KNN. Kết quả độ sai số trung bình từ 1,39 đến 1,5m tùy vào giá trị K và so sánh với các phương pháp DT, KNN, DNN, SVM, và RF thì mô hình kết hợp có kết quả tốt hơn.

Giải pháp dựa trên thuật toán phân lớp SVM đã được đề xuất trong[16, 17]. Kết quả của các phương pháp đều cho thấy SVM cho độ chính xác cao hơn thuật toán fingerPrinting. Độ chính xác là 2m trong 77% và 98,75% trong 93,75% các trường hợp kiểm thử.

LR được các tác giả sử dụng trong [12] sau khi tối ưu dữ liệu, kết quả độ chính xác đạt 95.83%. Chenlu Xiang và các cộng sự dùng LR kết hợp với tối ưu dữ liệu ở môi trường phòng thí nghiệm tiêu chuẩn trong [18, 19] đều cho kết quả sai số định vị đạt 92cm.

Các nghiên cứu sử dụng tập dữ liệu UJIIndoorLoc. Trong [20] các tác giả dùng KNN phân cụm dự báo vị trí tòa nhà và KNN hồi quy để dự báo tọa độ và RF để dự báo tòa-tầng. Sau khi loại bỏ các AP chỉ xuất hiện dưới 2 lần, kết quả cho thấy RF dự đoán vị trí tòa nhà đúng 100%, dự đoán tầng bằng RF đạt kết quả lần lượt 97.95%, 90.87% and 95.86% cho ba tòa nhà, kết quả dự đoán kinh độ vĩ độ ở ba tòa nhà có giá trị (RMSE) trong khoảng 7.1 đến 12.8, Mean Absolute Error (MAE) trong khoảng 5 đến 9.
Trong [21] sử dụng cách kết hợp hệ thống COSELM (Constraint Online Sequential Extreme Learning Machine) với KNN thành hệ thống có tên AFARLS. Kết quả AFARLS có hiệu suất dự đoán tầng 95.41%, sai số vị trí 6.4m, trong khi KNN dự đoán tầng đạt 89.92%. Trong [22], các tác giả đã biến đổi tập dữ liệu huấn luyện theo cách riêng của mình và tạo ra tập dữ liệu thử nghiệm mới từ tập dữ liệu đã biến đổi, trong đó 520 đặc trưng chỉ còn lại 60 đặc trưng. Thử nghiệm được thực thi bởi 6 thuật toán, trong đó thuật toán KNN nhận diện tầng đạt 97%.

Áp dụng lượng tử hóa trước khi áp dụng KNN, SVM, RF trong [23], kết quả nghiên cứu có độ chính xác vị trí tương ứng với các thuật toán KNN, SVM, RF là 67.49%, 62,71% và 68.5%, thuật toán RF dùng để ước lượng vị trí tầng đạt độ chính xác 97%.

Như vậy các thuật toán KNN, SVM, LR đã được nhiều nhóm nghiên cứu sử dụng bằng nhiều cách khách nhau. Theo kiến thức tốt nhất mà chúng tôi có, hiện tại trên tập dữ liệu UJIIndoorLoc chưa có nghiên cứu nào kết hợp giữa các thuật toán như mô hình chúng tôi đề xuất. Do đó trong bài báo này chúng tôi tiến hành xây dựng mô hình kết hợp tổ hợp các thuật toán KNN, SVM, LR cho phần huấn luyện mô hình ở giai đoạn huấn luyện và thực nghiệm trên bộ dữ liệu UJIIndoorLoc nhằm nâng cao hiệu suất định vị.

III. ĐỀ XUẤT MÔ HÌNH ĐỊNH VỊ

1. Mô tả tập dữ liệu

Tập dữ liệu UJIIndoorLoc[13] được nhiều nghiên cứu sử dụng [8], UJIIndoorLoc có số lượng RP là 933, các RP thuộc 3 tòa nhà cao 4 hoặc 5 tầng với tổng diện tích 108703m^2. UJIIndoorLoc có 21049 mẫu, trong đó 19938 mẫu dùng cho huấn luyện và 1111 mẫu dùng cho kiểm thử. Mỗi mẫu gồm 529 đặc trưng, 520 đặc trưng đầu tiên là giá trị RSS thu được từ 520 AP, giá trị RSS nằm trong

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

khoảng từ -104dB đến 0dB, giá trị RSS=100 được đặt mặc định với các AP không phát hiện được khi lấy mẫu. Chín đặc trưng còn lại là các nhãn gồm kinh độ (Longitude), vĩ độ (Latitude), số hiệu tòa nhà (BuildingID), tầng trong tòa nhà (Floor), số hiệu vùng lấy mẫu (SpaceID), vị trí tương quan xác định vùng lấy mẫu là trong hay ngoài các phòng (Relative Position), mã người thu thập dữ liệu (UserId), mã điện thoại thu thập dữ liệu (PhoneId) và thời gian thu thập (Timestamp).

2. Đề xuất kỹ thuật giảm kích thước dữ liệu cho bài toán phân lớp

Tập dữ liệu UJIIndoorLoc như mô tả bên trên là tập dữ liệu lớn, do đó cần tiến hành xử lý dữ liệu nhằm giảm thời gian tính toán và tăng chất lương định vị.

Trong UJIIndoorLoc, mỗi tòa nhà là một lớp bao gồm nhiều dòng dữ liệu, một dòng dữ liệu gốc có dạng $[RSS_1, RSS_2, \cdots, RSSI_{520}$, buildingID], trong đó 520 giá trị đầu tiên là các giá trị RSS đo được từ 520 AP, giá trị cuối cùng là số hiệu ba tòa nhà, các tòa nhà được đánh số từ 0 đến 2. Với mỗi tòa nhà chúng tôi lấy giá trị RSS trung bình của mỗi AP, kết quả thu được 520 giá trị trung bình. Kết quả thu được tập dữ liệu mới được thể hiện trong Phương trình (1) với 520 dữ liệu đầu là giá trị mới của các đặc trưng và cuối cùng là số hiệu tòa nhà.

$$\begin{pmatrix} \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 0 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 1 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 2 \end{pmatrix} (1)$$

Bài toán phân lớp giá trị giữa các điểm càng chênh lệch lớn thì càng dễ phân lớp, chẳng hạn như tòa nhà được phân lớp tốt hơn nếu các đặc trưng có nhiều thông tin của tòa nhà. Do đó, chúng tôi tìm sự khác biệt tuyệt đối giữa các cặp tòa nhà 0-1, 0-2 và 1-2 bằng cách ghép chéo các cặp tương ứng trong tập dữ liệu (1), chúng tôi thu được ba mảng tương ứng với các cặp tòa nhà. Sắp xếp ba mảng thu được theo thứ tự giảm dần các đặc trưng và loại bỏ các đặc trưng có sự khác biệt thấp, trùng nhau. Kết quả mỗi mảng còn lại N đặc trưng có sự khác biệt cao nhất. Hợp nhất ba mảng lại thành một tập hợp đặc trưng có N*3 giá trị cho mô hình phân lớp tòa nhà. Phương pháp giảm kích thước dữ liệu tòa nhà thể hiện trong Thuật toán 1.

Tập dữ liệu dùng phân lớp tầng được chúng tôi xử lý tương tự như tòa nhà. Một dòng dữ liệu gốc có dạng $[RSS_1, RSS_2, \cdots, RSSI_{520}$, floor], trong đó 520 giá trị đầu tiên là các giá trị RSS đo được từ 520 AP, giá trị cuối cùng là số hiệu các tầng, các tầng được đánh số từ 0 đến 4, kết quả thể hiện theo Phương trình (2).

Thuật toán 1: Thuật toán giảm kích thước đặc trưng

Dữ liệu vào:
$\mathcal{B}_0 \leftarrow$:các dòng dữ liệu của buildingID=0;
$\mathcal{B}_1 \leftarrow$:các dòng dữ liệu của buildingID=1;
$\mathcal{B}_2 \leftarrow$:các dòng dữ liệu của buildingID=2;
Dữ liệu ra: $\mathcal{B}$: Tập hợp các đặc trưng đã rút gọn.
begin
  Bước 1:
  1.1. $\mathcal{BA}_0 \leftarrow$ Tập giá trị RSS trung bình của $\mathcal{B}_0$;
  1.2. $\mathcal{BA}_1 \leftarrow$ Tập giá trị RSS trung bình của $\mathcal{B}_1$;
  1.3. $\mathcal{BA}_2 \leftarrow$ Tập giá trị RSS trung bình của $\mathcal{B}_2$;
  Bước 2:
  2.1. $\mathcal{B}_{01} \leftarrow \mathcal{BA}_0 \cup \mathcal{BA}_1$;
  2.2. $\mathcal{B}_{02} \leftarrow \mathcal{BA}_0 \cup \mathcal{BA}_2$;
  2.3. $\mathcal{B}_{12} \leftarrow \mathcal{BA}_1 \cup \mathcal{BA}_2$;
  Bước 3: Sắp xếp $\mathcal{B}_{01}$, $\mathcal{B}_{02}$ và $\mathcal{B}_{12}$ theo thứ tự giảm dần;
  Bước 4: Loại bỏ các đặc trưng có sự khác biệt thấp, trùng nhau;
  Bước 5: $\mathcal{B} \leftarrow \mathcal{B}_{01} \cup \mathcal{B}_{02} \cup \mathcal{B}_{12}$;
end

$$\begin{pmatrix} \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 0 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 1 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 2 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 3 \\ \overline{RSS_1}, & \overline{RSS_2}, & \cdots, & \overline{RSSI_{520}}, & 4 \end{pmatrix} (2)$$

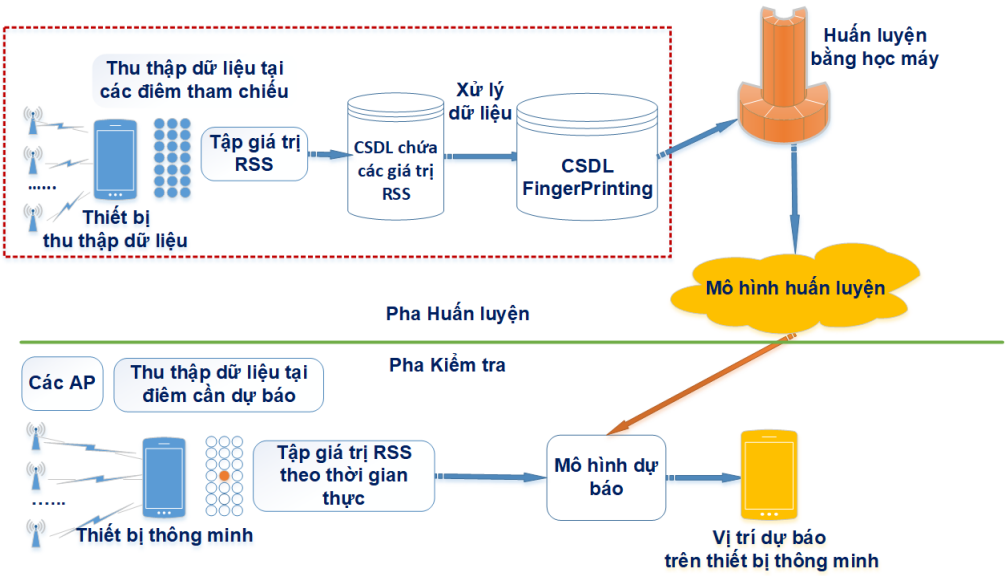
Bài toán phân lớp tầng tìm các đặc trưng nhiều thông tin tương tự, dữ liệu phân lớp tầng được xử lý tương tự như dữ liệu tòa nhà. Chúng tôi kết hợp các cặp tầng 0-1, 0-2, 0-3, 0-4, 1-2, 1-3, 1-4, 2-3, 2-4, 3-4 tạo thành 10 mảng. Sau khi loại bỏ các đặc trưng có độ chênh lệch thấp bằng phương pháp đã áp dụng cho tòa nhà, chúng tôi gộp các mảng lại để tạo thành tập dữ liệu đầu vào có 10*N đặc trưng cho mô hình phân lớp tầng.

Cuối cùng, gộp các tập dữ liệu lại với nhau, chúng tôi thu được 13 lớp cho các tầng của các tòa nhà. Giá trị N được tính theo phương pháp heuristically nó ảnh hưởng trực tiếp đến hiệu suất của thuật toán.

3. Mô hình đề xuất

Mô hình cơ bản của bài toán định vị trong nhà bằng fingerprinting dùng sóng WiFi dựa trên học máy được thể hiện trong Hình 1. Trong mô hình, thuật toán học máy sẽ được áp dụng trên tập dữ liệu fingerPrinting (tập dữ liệu huấn luyện) để đưa ra mô hình huấn luyện tại giai đoạn huấn luyện. Ở giai đoạn kiểm thử, tập RSS thời gian thực được đưa vào mô hình dự báo để tìm ra vị trí ước lượng theo mô hình huấn luyện.

Trong bài báo này chúng tôi đề xuất mô hình học máy mới, mô hình này dùng các thuật toán phân lớp nhằm ước



Hình 1. Mô hình định vị trong nhà cơ bản dựa trên học máy

lượng vị trí theo tòa-tầng. Khác với mô hình cơ bản, mô hình của chúng tôi tích hợp nhiều thuật toán và chia làm hai pha như Hình 2. Pha đầu tiên, tập dữ liệu fingerprinting được huấn luyện bằng tổ hợp các thuật toán phân lớp LR, KNN và SV. Pha một tạo ra tổ hợp dữ liệu mới, tổ hợp dữ liệu này là đầu vào cho thuật toán huấn luyện LR ở pha thứ hai. Như vậy, mô hình của chúng tôi huấn luyện tập dữ liệu theo hai pha liên tiếp thay vì một pha như mô hình cơ bản. Hình 3 mô tả chi tiết hoạt động của tổ hợp thuật toán theo hai pha. Trong đó $\hat{Y}_1$ $\hat{Y}_2$ và $\hat{Y}_3$ là kết quả huấn luyện của ba thuật toán trong pha một, $\hat{Y}_f$ là kết quả của pha hai và cũng là mô hình huấn luyện cuối.

IV. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

1. Tiền xử lý dữ liệu

Chúng tôi phân tích dữ liệu UJIIndoorLoc và chúng tôi nhận thấy cần tiền xử lý dữ liệu như sau: thứ nhất số lượng các AP nhận được tại một RP nhỏ nhất là 0, lớn nhất là 51 và trung bình là xấp xỉ 18 AP, trong khi bài toán định vị trong nhà chỉ cần tối thiểu 3 AP là có thể xác định được vị trí và theo thống kê thì số RP có AP nhỏ hơn 6 (bao gồm cả các RP không có AP nào) là 389, do đó chúng tôi quyết định loại bỏ các RP có AP nhỏ hơn 6, điều này hầu như không ảnh hưởng đến chất lượng định vị nhưng lại tăng tốc độ tính toán. Thứ hai tại một RP có thể nhận được giá trị RSS từ AP của tòa nhà khác, các AP này không có tác dụng cho định vị thậm chí làm giảm chất lượng, do đó chúng tôi cũng loại bỏ các AP này. Tiền xử lý dữ liệu được áp dụng cho pha kiểm thử của mô hình phân lớp.

2. Các chỉ số đánh giá

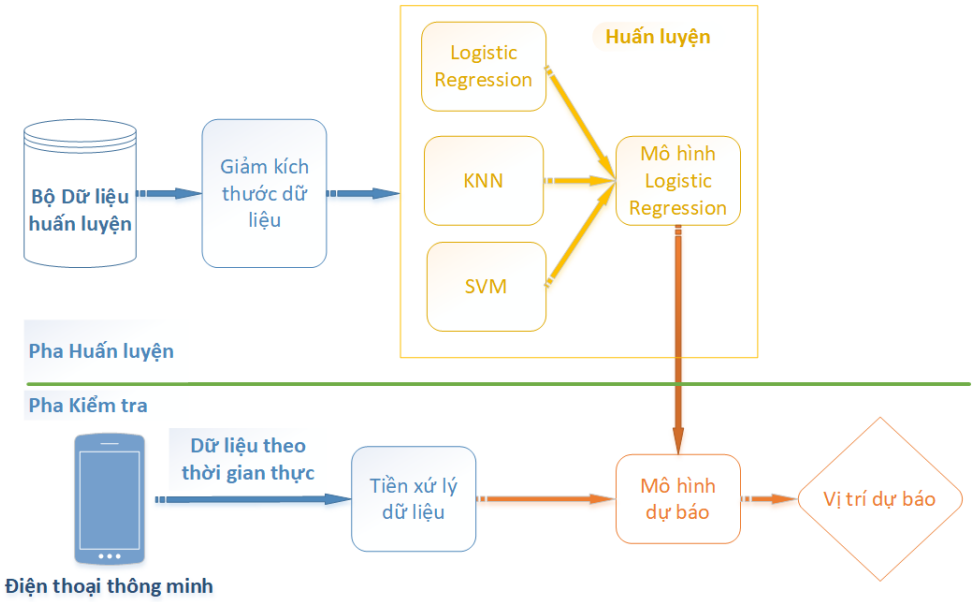
Bài toán phân lớp có các chỉ số đánh giá dựa trên ma trận nhầm lẫn (confusion matrix) để đếm được số lượng các giá trị gồm True Positive (TP): số lượng điểm của lớp positive được phân loại đúng là positive; True Negative (TN): số lượng điểm của lớp negative được phân loại đúng là negative; False Positive (FP): số lượng điểm của lớp negative bị phân loại nhầm thành positive và False Negative (FN): số lượng điểm của lớp positive bị phân loại nhầm thành negativ. Gọi T là tổng số các điểm của cả hai lớp khi đó các chỉ số đánh giá phân lớp bao gồm:

- Accuracy: Độ chính xác $Accuracy = \frac{TP+TN}{T}$
- Precision:tỉ lệ số điểm true positive trong số những điểm được phân loại là positive. $Precision = \frac{TP}{TP+FP}$
- Recall: tỉ lệ số điểm true positive trong số những điểm thực sự là positive. $ReCall = \frac{TP}{TP+FN}$
- F1-Score: trung bình điều hòa (harmonic mean) của các tiêu chí Precision và Recall, F1-Score đánh giá khách quan khả năng thực thi của mô hình học máy. $F_1 = 2\frac{Precision.Recall}{Precision+Recall}$

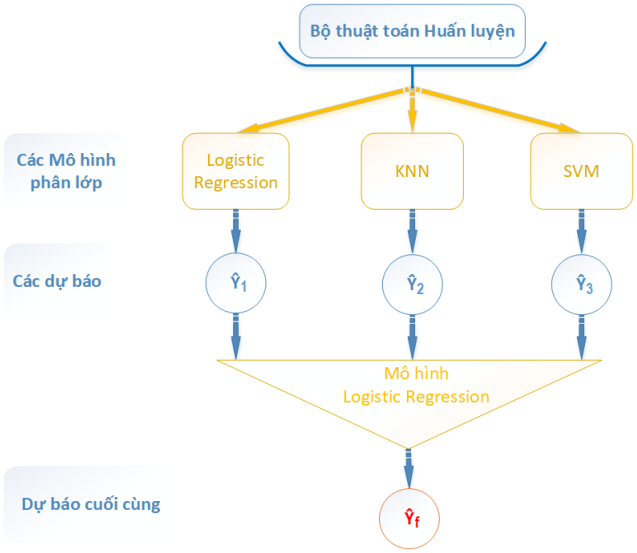
3. Kết quả thực nghiệm

Trong phần này chúng tôi cung cấp các kết quả thực nghiệm thể hiện hiệu quả của các thuật toán độc lập và phương pháp kết hợp mà chúng tôi đề xuất, kết quả cho thấy khi tích hợp các thuật toán kết quả tốt hơn khi sử dụng từng thuật toán độc lập

Mô hình kết hợp dự đoán tòa-tầng được chúng tôi thực hiện theo hai bước, đầu tiên chúng tôi thử nghiệm các mô hình dùng thuật toán phân lớp độc lập nhằm lựa chọn các



Hình 2. mô hình kết hợp phân loại tòa-tầng



Hình 3. Mô hình hoạt động tổ hợp thuật toán phân loại tòa-tầng

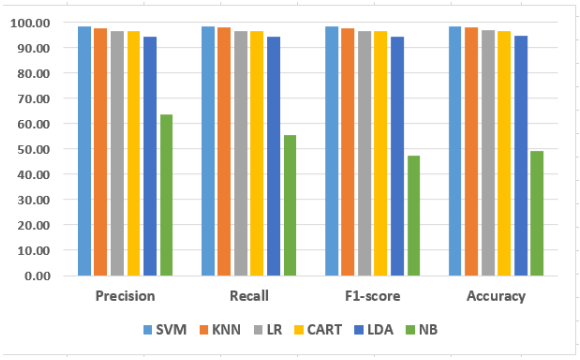
thuật toán tốt nhất. Các thuật toán chúng tôi thử nghiệm bao gồm KNN, SVM, LR, Naïve Bayes (NB), cây phân loại và hồi quy (Classification and Regression Tree-CART) và Linear Discriminant Analysis (LDA).

Các kết quả của các thuật toán độc lập được tổng hợp trong Bảng I, giá trị các chỉ số được so sánh ở biểu đồ trong Hình 4 và hình ảnh của Hình 5. Bảng I và các hình cho thấy tất cả các chỉ số của các thuật toán LR, KNN, SVM đều cho kết quả tốt hơn các thuật toán còn lại. Lưu ý, chỉ số của LR và CART rất gần nhau, với bài toán phân lớp chúng tôi chỉ chọn một, chỉ số của Recall của thuật toán LR thấp hơn của thuật toán CART, tuy nhiên chỉ số F1-Score của LR lại tốt hơn CART, cho thấy mô hình của LR tốt hơn nên chúng tôi chọn LR. Dùng các kết quả này, chúng tôi chọn 3 thuật toán có kết quả tốt nhất LR, KNN và SVM vào trong mô hình kết hợp.

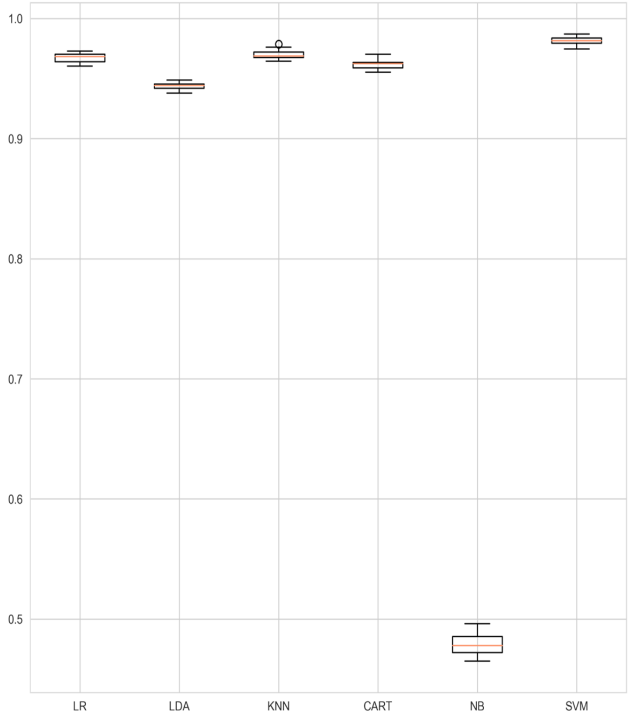
Bước hai, chúng tôi thử nghiệm mô hình kết hợp các thuật toán đã được chọn. Trong Bảng II là kết quả tổng hợp của các chỉ số Precision, Recall, F1-score và Accuracy của mô hình kết hợp Kết quả của mô hình kết hợp so với các mô hình sử dụng các thuật toán độc lập được thể hiện ở biểu đồ trong Hình 6 cho thấy tất cả các chỉ số của mô

Bảng I
TỔNG HỢP KẾT QUẢ CÁC CHỈ SỐ
CỦA CÁC THUẬT TOÁN ĐỘC LẬP

	SVM	KNN	LR	CART	LDA	NB
Precision	98.43	97.71	96.62	96.50	94.42	63.70
Recall	98.47	97.98	96.69	96.71	94.26	55.37
F1-score	98.45	97.83	96.65	96.60	94.33	47.42
Accuracy	98.57	97.93	96.86	96.76	94.66	49.09
Time (s)	7.95	0.04	3.19	0.47	1.21	0.67



Hình 4. Biểu đồ so sánh giá trị tổng hợp các chỉ số các mô hình phân lớp độc lập



Hình 5. Biểu đồ so sánh độ chính xác của các mô hình phân lớp độc lập

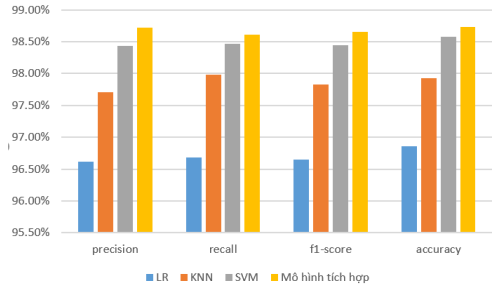
hình kết hợp có kết quả tốt hơn kết quả của tất cả các thuật

Bảng II
KẾT QUẢ ƯỚC LƯỢNG TÒA-TẦNG CỦA MÔ HÌNH KẾT HỢP

	Pre-cision	Re-call	F1-Score	Accu-racy	Time (s)
Mô hình kết hợp	98.71	98.61	98.66	98.73	99.31

toán LR, KNN và SVM.

Như vậy, mô hình kết hợp các thuật toán của chúng tôi đề xuất đạt hiệu suất và kết quả tốt hơn khi dùng các thuật toán độc lập.



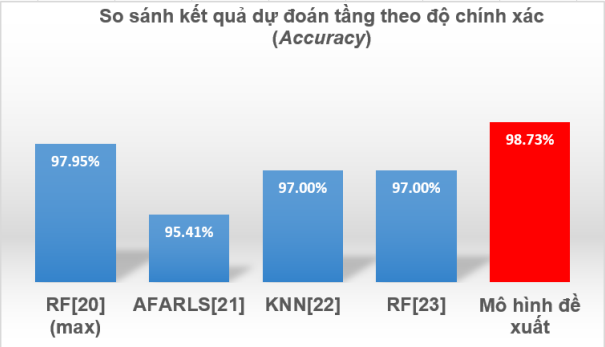
Hình 6. So sánh các chỉ số phân lớp giữa mô hình kết hợp và các mô hình độc lập

4. So sánh kết quả với các nghiên cứu khác trên cùng tập dữ liệu UJIIndoorLoc

Hình 7 thể hiện kết quả dự đoán tòa-tầng của mô hình kết hợp mà chúng tôi đề xuất với các nghiên cứu khác, trong đó các nghiên cứu khác được thể hiện bằng tên thuật toán hoặc tên do nhóm nghiên cứu đặt cùng với số tham chiếu của nghiên cứu. Với nghiên cứu của chúng tôi, nhiều đơn vị đo được sử dụng để đánh giá kết hiệu suất mô hình cũng như độ chính xác, tuy nhiên, các nghiên cứu khác thông thường chỉ công bố độ chính xác, do đó, trong so sánh chúng tôi dùng chỉ số độ chính xác (Accuracy) để so sánh, ngoài ra với các công bố không sử dụng kết quả trung bình, chúng tôi sử dụng kết quả cao nhất mà nhóm công bố (ghi chú max). Kết quả cho thấy mô hình của chúng tôi dự đoán tốt hơn các mô hình khác sử dụng cùng các thuật toán.

V. KẾT LUẬN

Trong bài báo này, phương pháp giảm kích thước dữ liệu được đề xuất bằng cách phân tích theo cặp giữa các tòa nhà, giữa các tầng đã mang lại kết quả rất tốt cho các mô hình phân lớp độc lập và kết hợp. Kết quả mô hình phân lớp kết hợp có độ chính xác cao 98.73% và cao hơn so với kết quả của các mô hình dùng các thuật toán độc lập. Giá trị của các chỉ số Precision, Recall rất cao thể hiện hiệu



Hình 7. So sánh kết quả dự đoán tòa-tầng theo độ chính xác (Accuracy) của mô hình kết hợp với các nghiên cứu khác

suất cao của mô hình, giá trị F1-Score cao khẳng định độ ổn định của hệ thống, các kết quả này cho thấy mô hình của chúng tôi đã thành công và cho kết quả tốt. So sánh kết quả với các nghiên cứu khác trên cùng tập dữ liệu, mô hình cũng cho kết quả tốt hơn. Tuy nhiên, mô hình vẫn có nhược điểm là thời gian chạy cao do tích hợp nhiều thuật toán, tuy nhiên mô hình được áp dụng ở pha huấn luyện mô hình nên thời gian chạy lâu không phải là vấn đề lớn.

Mô hình kết hợp có rất nhiều hứa hẹn áp dụng cho các bài toán khác và chúng tôi có thể nâng cấp được mô hình khi thay thế bằng các thuật toán học sâu.

TÀI LIỆU THAM KHẢO

[1] S. Chan and G. Sohn, "Indoor localization using wi-fi based fingerprinting and trilateration techiques for lbs applications," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, no. 4, p. C26, 2012.

[2] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1281–1293, 2013.

[3] C. Basri and A. El Khadimi, "Survey on indoor localization system and recent advances of wifi fingerprinting technique," in *2016 5th international conference on multimedia computing and systems (ICMCS)*. IEEE, 2016, pp. 253–259.

[4] E. Laitinen, E. S. Lohan, J. Talvitie, and S. Shrestha, "Access point significance measures in wlan-based location," in *2012 9th Workshop on Positioning, Navigation and Communication*, 2012, pp. 24–29.

[5] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, 2016.

[6] D. Chu, L.-Z. Liao, M. K. Ng, and X. Wang, "Incremental linear discriminant analysis: A fast algorithm and comparisons," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp. 2716–2735, 2015.

[7] A. Nessa, B. Adhikari, F. Hussain, and X. N. Fernando, "A survey of machine learning for indoor positioning," *IEEE Access*, vol. 8, pp. 214 945–214 965, 2020.

[8] N. Singh, S. Choe, and R. Punmiya, "Machine learning based indoor localization using wi-fi rssi fingerprints: an overview," *IEEE Access*, 2021.

[9] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on computer communications. Nineteenth annual joint conference of the IEEE computer and communications societies (Cat. No. 00CH37064)*, vol. 2. Ieee, 2000, pp. 775–784.

[10] M. Y. Umair, K. V. Ramana, and D. Yang, "An enhanced k-nearest neighbor algorithm for indoor positioning systems in a wlan," *2014 IEEE Computers, Communications and IT Applications Conference*, pp. 19–23, 2014.

[11] C. Sujin and G. Kousalya, *Fingerprint-Based Support Vector Machine for Indoor Positioning System*, 01 2019, pp. 289–298.

[12] Z. Peng, Y. Xie, D. Wang, and Z. Dong, "One-to-all regularized logistic regression-based classification for wifi indoor localization," *2016 IEEE 37th Sarnoff Symposium*, pp. 154–159, 2016.

[13] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, and J. Huerta, "Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems," in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2014, pp. 261–270.

[14] M. T. Hoang, Y. Zhu, B. Yuen, T. Reese, X. Dong, T. Lu, R. Westendorp, and M. Xie, "A soft range limited k-nearest neighbors algorithm for indoor localization enhancement," *IEEE Sensors Journal*, vol. 18, no. 24, pp. 10 208–10 216, 2018.

[15] P. Dai, Y. Yang, M. Wang, and R. Yan, "Combination of dnn and improved knn for indoor location fingerprinting," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.

[16] L. Zhang, Y. Li, Y. Gu, and W. Yang, "An efficient machine learning approach for indoor localization," *China Communications*, vol. 14, no. 11, pp. 141–150, 2017.

[17] Y. Rezgui, L. Pei, X. Chen, F. Wen, and C. Han, "An efficient normalized rank based svm for room level indoor wifi localization with diverse devices," *Mobile Information Systems*, vol. 2017, pp. 1–19, 07 2017.

[18] C. Xiang, Z. Zhang, S. Zhang, S. Xu, S. Cao, and V. K. N. Lau, "Robust sub-meter level indoor localization - a logistic regression approach," *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2019.

[19] C. Xiang, S. Zhang, S. Xu, X. Chen, S. Cao, G. C. Alexandropoulos, and V. K. N. Lau, "Robust sub-meter level indoor localization with a single wifi access point—regression versus classification," *IEEE Access*, vol. 7, pp. 146 309–146 321, 2019.

[20] P. R. Shivam Wadhwa and R. Kaushik, "Machine learning based indoor localization using wi-fi fingerprinting," *International Journal of Recent Technology and Engineering*, 2019.

[21] H. Gan, M. H. B. M. Khir, G. Witjaksono Bin Djaswadi, and N. Ramli, "A hybrid model based on constraint oselm, adaptive weighted src and knn for large-scale indoor localization," *IEEE Access*, vol. 7, pp. 6971–6989, 2019.

[22] G. H. Apostolo, I. G. B. Sampaio, and J. Viterbo, "Feature selection on database optimization for wi-fi fingerprint indoor positioning," *Procedia Computer Science*, vol. 159, pp. 251–260, 2019, knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 23rd International Conference KES2019.

[23] W. Charoenruengkit, S. Saejun, R. Jongfungfeuang, and K. Multhonggad, "Position quantization approach with multi-class classification for wi-fi indoor positioning system," in *2018 International Conference on Information Technology (InCIT)*, 2018, pp. 1–5.

SƠ LƯỢC VỀ TÁC GIẢ



Ngô Văn Bình Nghiên cứu sinh tại Học Viện Khoa học Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam. Nhận bằng Cử nhân khoa học ngành Tin học năm 1999 tại Trường Đại học Khoa học Tự nhiên- Đại học Quốc gia Hà Nội. Nhận bằng Thạc sĩ năm 2004 tại Khoa Công nghệ - Đại học Quốc gia Hà Nội.
Hiện nay đang giảng dạy tại Đại học FPT.
Lĩnh vực nghiên cứu: Định vị trong nhà, học máy.
Email: binhnv11@fe.edu.vn



Vũ Văn Hiệu Nhận bằng tiến sĩ Cơ sở toán học cho tin học tại Học Viện Khoa học Công nghệ, Viện hàn lâm Khoa học và Công nghệ Việt Nam năm 2017. Nhận bằng thạc sĩ Khoa học máy tính vào năm 2007 và Kỹ sư Công nghệ thông tin vào năm 2003.
Hiện nay đang nghiên cứu tại Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.
Lĩnh vực nghiên cứu: Thị giác máy tính, khoa học dữ liệu.
Email: vvhieu@ioit.ac.vn

Hệ gợi ý mua sắm dựa theo phiên làm việc với mô hình mạng học sâu đồ thị

Nguyễn Tuấn Khang[1], Nguyễn Tú Anh[2], Mai Thúy Nga[2], Nguyễn Hải An[3], Nguyễn Việt Anh[1]
[1] Viện Công Nghệ Thông Tin, Học Viện Khoa Học Công Nghệ, Hà Nội
[2] Trường Đại học Thăng Long, Hà Nội
[3] Phòng Khoa Học Công Nghệ, Tổng Công Ty Thăm Dò Khai Thác Dầu Khí, Hà Nội

Tác giả liên hệ: Nguyễn Tuấn Khang, khang_nt@yahoo.com
Ngày nhận bài: 30/08/2022, ngày sửa chữa: 13/11/2022, ngày duyệt đăng: 25/11/2022
Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1135

Tóm tắt: Phân tích phiên làm việc của khách hàng để dự báo khả năng họ sẽ lựa chọn sản phẩm nào tiếp theo là một bài toán dự báo khá phổ biến trong ngành thương mại điện tử. Việc dự báo này giúp cho doanh nghiệp đưa ra các ý tưởng bán hàng phù hợp trong quá trình người dùng tương tác với hệ thống bán hàng của mình. Bài báo này đề xuất hướng sử dụng mạng học sâu đồ thị để xây dựng mô hình gợi ý dựa vào phiên làm việc của khách hàng. Kết quả thực nghiệm cho thấy việc sử dụng đồ thị rất phù hợp trong việc biểu diễn dữ liệu lựa chọn sản phẩm thông qua hành vi nhấp chuột trong phiên làm việc của khách hàng và mô hình gợi ý sử dụng GNN cho kết quả dự báo với 2 chỉ số đánh giá mô hình Recall@20 và MRR@20 tốt hơn so với các mô hình trước đây.

Từ khóa: *dữ liệu nhấp chuột, hành vi mua sắm, hệ thống gợi ý, phiên làm việc, mạng học sâu đồ thị (Graph Neural Network).*

Title: **Session-based Recommendation using Graph Neural Network**

Abstract: Customer behavior analysis based on the currrent active session to understand and predict what is the next product that customer might click is a potential usecase in ecommerce. This type of recommendation helps enterprise to promote the upselling opportunity to increase the purchase behavior. This paper proposes to use a GNN to develop a recommendation model using the current active session of customer during their purchasing clicks on the website. The experimental result shows that the GNN is suitable to model a product selection from sequential mouse clicks, and the session-based recommendation using GNN performs higher than other models with the two performance metrics Recall@20 and MRR@20.

Keywords: *mouse click, purchase behaviour, recommdendation system, session, graph neural network (GNN)*

I. TỔNG QUAN

1. Tổng quan bài toán

Khi một khách hàng vào một trang thương mại điện tử thì có hai xu hướng: hoặc họ đã định hướng được sản phẩm mà họ sẽ mua, hoặc là họ được định hướng được sản phẩm mà họ nên mua. Đối với kịch bản thứ hai, người dùng sẽ gặp khó khăn hơn nhiều vì họ sẽ phải chọn sản phẩm phù hợp nhất với nhu cầu của họ. Vấn đề đặt ra là làm sao họ có thể làm được điều đó trong vô số sản phẩm giống nhau mà họ đang tìm kiếm, đó chính là ý tưởng xây dựng hệ thống gợi ý [1].

Các hệ thống gợi ý ngày nay càng được chú trọng, nhất là đối với các nhà cung cấp dịch vụ trực tuyến như: Amazon, Netflix [2], Youtube... Một hệ thống gợi ý hiệu quả sẽ là vấn đề sống còn đối với nhà cung cấp dịch vụ hoặc bán hàng, nó có thể làm tăng sự hài lòng của khách hàng và giữ chân người dùng lâu dài [3].

Hiện này có hai hướng để xây dựng hệ thống gợi ý tùy theo ngữ cảnh tương tác người dùng như sau:

- Hệ gợi ý dựa vào thông tin lịch sử hoặc sở thích của người dùng đã để lại để tìm ra sản phẩm phù hợp nhất, hệ thống hoạt động kiểu này khá dễ hiểu nhưng lại gặp nhiều thách thức khi cần đưa ra gợi ý cho người dùng ngay cả khi họ không để lại thông tin lịch sử gì cho hệ thống.
- Hệ gợi ý dựa chỉ dựa vào quá trình tương tác hiện tại của người dùng với hệ thống, gọi là phiên làm việc, nhằm cho phép hệ thống có thể đưa ra gợi ý cho người

dùng chỉ sau vài ba chuỗi sự kiện tương tác của họ với hệ thống, mô hình này được gọi là hệ thống gợi ý dựa vào phiên làm việc (*Session-based Recommendation*), gọi tắt là bài toán SR.

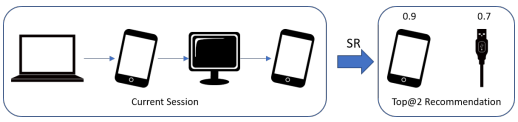
2. Đặt vấn đề

Đối tượng nghiên cứu của bài báo này là hành vi nhấp chuột (lựa chọn sản phẩm) của khách hàng trong một phiên mua hàng. Mục tiêu của bài báo này là nghiên cứu và đề xuất mô hình dự báo hành vi lựa chọn sản phẩm của khách hàng mô hình mạng học sâu, cụ thể hơn là trả lời câu hỏi *"Với số lượng sản phẩm đã lựa chọn trong phiên tương tác hiện tại thì khả năng người dùng sẽ chọn sản phẩm nào tiếp theo"*. Ở mức độ tổng quan, mô hình gợi ý sẽ không chỉ đưa ra một sản phẩm tiếp theo mà sẽ đưa ra danh sách gợi ý K sản phẩm có xác xuất cao nhất mà khách hàng có thể lựa chọn một, bài toán này còn gọi là bài toán gợi ý Top-K, ngắn gọn gọi là bài toán Top-K.

Bài toán SR có thể được mô tả như sau, giả sử $V = \{v_1, v_2, ..., v_m\}$ là một danh mục các đối tượng duy nhất (ví dụ như danh mục sản phẩm) được các người dùng tương tác trong các phiên làm việc của họ, như vậy ta có thực thể phiên làm việc được biểu diễn như sau $s = [v_{s,1}, v_{s,2}, ..., v_{s,n}]$ trong đó $v_{s,i} \in V$ có tính thứ tự theo chuỗi thời gian để và thể hiện một hành động lựa chọn (click) nào đó của người dùng trong phiên làm việc s (ví dụ chọn một sản phẩm cụ thể).

Để giải quyết vấn đề này, ta cần xây dựng mô hình dự báo liệu người dùng sẽ lựa chọn đối tượng (sản phẩm) $v_{s,n+1}$ tiếp theo trong phiên làm việc s đó. Với mô hình gợi ý này cho một phiên làm việc s cụ thể, hệ gợi ý sẽ trả về hàm $\hat{y}$ là một véc-tơ chứa danh mục k sản phẩm gợi ý nào đó với xác xuất được lựa chọn từ cao tới thấp, danh mục sản phẩm gợi ý này được gọi là $top - k$ sản phẩm gợi ý cho người dùng.

Hình 1 minh họa mô hình SR đưa ra dự báo $top - 2$ sản phẩm cùng xác xuất tương ứng mà khách hàng sẽ lựa chọn để click tiếp.



Hình 1. Bài toán gợi ý top-k sản phẩm

II. NGHIÊN CỨU LIÊN QUAN

Bài toàn gợi ý trong lĩnh vực thương mại điện tử không phải là vấn đề mới, ngay từ những năm 2000 JB Schafer và cộng sự [1] đã nêu ra vấn đề này để tìm cách nâng cao khả năng bán kèm và bán chéo sản phẩm cho các website bán hàng thời đó, thuật toán được đề xuất là sử dụng các thông tin trong quá khứ cũng như sở thích cá nhân (*personalization*) của khách hàng đề để xuất sản phẩm cần bán trong tương lai. Sau đó, nhóm tác giả này tiếp tục cải thiện mô hình gợi ý với ý tưởng sử dụng dữ liệu tri thức và sự tương quan (*correlation*) giữa các sản phẩm hay giữa các người dùng để phân tích hành vi khách hàng (*customer behavior*) [4] với việc phân tích các hệ thống thương mại điện tử thời đó như amazon hay ebay.

Badrul M. Sarwar và cộng sự (2000) [5] nhận thấy các hệ thống gợi ý đang phải xử lý khối lượng thông tin khổng lồ về sản phẩm, khách hàng, đơn hàng gây ra thách thức trong việc đưa ra gợi ý, nhóm tác giả đề xuất một trong những thuật toán phổ biến của học máy là SVD (*Singular Value Decomposition*) với mục tiêu giảm số chiều thông tin để tăng tốc độ xử lý của hệ thống gợi ý, nghiên cứu này cũng đưa ra khái niệm "*top N list*" của mô hình gợi ý. Năm 2002, nhóm tác giả này tiếp tục đề xuất sử dụng thuật toán người láng giềng (*neighborhood*) [6] để phân nhóm khác hàng (*clustering*) từ đó xây dựng mô hình gợi ý theo mô hình bộ lọc cộng tác (*collaborative filtering*).

Năm 2004, Zan Huang và cộng sự [7] đưa khái niệm đồ thị vào bài tới gợi ý trong lĩnh vực thương mại điện tử, nhóm tác giả đề xuất đồ thị có hướng gồm hai lớp: lớp sản phẩm và lớp khách hàng mà mối quan hệ giữa hai lớp thể hiện thông tin mua sắm trong quá khứ thông qua trọng số của đồ thị. Mô hình đồ thị được thực nghiệm với cả ba kỹ thuật gợi ý gồm sử dụng bộ lọc cộng tác, dựa vào nội dung và hướng kết hợp, kết quả cho thấy mô hình này hoạt động tốt nhất với kỹ thuật kết hợp.

Năm 2006, Netflix tổ chức cuộc thi tìm kiếm giải thuật gợi ý tốt nhất nhằm dự đoán điểm đánh giá của người dùng cho các bộ phim của họ (*user ratings*) dựa vào các đánh giá trước đây mà không sử dụng thêm thông tin gì về người dùng hay bộ phim, đây là bài toán gợi ý lọc cộng tác. Yehuda Koren, Robert Bell và Chris Volinsky là thành viên đội thắng cuộc năm 2019 [2] trình bày mô hình phân tích ma trận thành nhân tử (*matrix factorization*) hoạt động tốt hơn các thuật toán của đối thủ khác như thuật toán SVD hoặc người láng giềng. Mô hình phân tích mà trận thành nhân tử tìm cách phân rã hai véc tơ đại diện cho người dùng (p_u) và bộ phim (q_i) (còn được gọi là véc tơ nhân tử, *factor vector*) vào một không gian nhân tử riêng (*joint-latent-factor space*), và vấn đề của mô hình là sao cho học được véc tơ nhân tử p_u và q_i với sai số *RMSE* (*root-mean-square error*) nhỏ nhất.

Balázs Hidasi và cộng sự (2015) [8] đưa ra mô hình mạng hồi quy RNN trong việc xây dựng hệ gợi ý, khác với bài toán của Netflix cần thông thông tin quá khứ của người dùng, hướng tiếp cận của nghiên cứu này tập

trung vào những phiên làm việc ngắn và hiện tại để đưa ra gợi ý cho người dùng, đó là khái niệm *Session-based Recommendation*. Mô hình này sử dụng thuật toán RNN phân cấp để tìm ra các đặc trưng ẩn trong phiên làm việc hiện tại của người dùng để đưa ra gợi ý cho sản phẩm tiếp theo, thuật toán RNN rất phù hợp với bài toán khi phải xử lý chuỗi sự kiện tuần tự (ví dụ như chuỗi nhấp chuột của người dùng trong phiên làm việc đó khi lựa chọn mua một số sản phẩm nào đó), nghiên cứu cho thấy mô hình RNN với biến thể HRNN (*Hierarchical RNN*) hoạt động tốt hơn so với các mô hình truyền thống cũng như mô hình RNN cơ sở.

Kế thừa nghiên cứu của Hidasi, Yong Kiam Tan và cộng sự (2016) [9] đã đề xuất cải tiến mô hình RNN với thuật toán xử lý dữ liệu làm việc theo chuỗi (phiên) phù hợp hơn cho mô hình RNN. Y.K. Tan và cộng sự cũng sử dụng chung bộ dữ liệu của Hidasi nhưng đã thực nghiệm đa dạng hơn để phân tích mức độ hiệu quả của việc xử lý dữ liệu phiên làm việc với mô hình RNN. Kết quả cho thấy nghiên cứu của Y.K. Tan và cộng sự cho kết quả tốt hơn và thuật toán xử lý dữ liệu được sử dụng tham chiếu trong một số nghiên cứu tiếp theo về bài toàn này [10–12]

Với sự ra đời của mô hình mạng học sâu và rộng (*Wide & Deep Learning*) do Google phát triển năm 2016, Heng-Tze Cheng và cộng sự [13], Khang Nguyen và cộng sự [14] cũng áp dụng mô hình này trong việc cải thiện tính tương tác giữa các thuộc tính ở cả mức cao và mức thấp để giúp cho mô hình gợi ý có thể tìm ra được các đặc tính ẩn tốt hơn do nó vừa có tính tổng quát hóa của mô hình học rộng vừa có tính ghi nhớ của mô hình học sâu.

Shu Wu và cộng sự (2019) [12] sử dụng mô hình mạng học sâu và đồ thị cùng khá nhiều kỹ thuật xử lý đồ thị và các biến thể khác nhau của GNN để phân tích bài toán gợi ý dựa vào phiên làm việc. Nhóm tác giả đề xuất kỹ thuật biến đổi véc tơ phiên làm việc sang một không gian nhúng bằng cách sử dụng mạng GNN để huấn luyện và học được véc tơ nhúng của đồ thị biểu diễn phiên làm việc, các véc tơ nhúng này thể hiện được các đặc tính ẩn của từng phiên làm việc từ đó hỗ trợ đưa ra được gợi ý có tính chính xác hơn.

III. PHƯƠNG PHÁP LUẬN

1. Ý tưởng xây dựng đồ thị

Ý tưởng của nghiên cứu là đề xuất một số phương án xây dựng đồ thị từ bộ dữ liệu lựa chọn sản phẩm của người mua hàng. Gọi d là số lượng sản phẩm trong bộ dữ liệu và các sản phẩm được đánh dấu theo số thứ tự từ 0 đến $d-1$.

Cụ thể nhóm tác giả đề xuất 3 dạng đồ thị sau:

- Đồ thị G: đồ thị đơn thể hiện mối quan hệ trực tiếp khi lựa chọn sản phẩm.
- Đồ thị H: đồ thị đơn thể hiện cả mối quan hệ trực tiếp và gián tiếp khi lựa chọn các sản phẩm trong phiên làm việc.
- Đồ thị K: đồ thị đa quan hệ (còn gọi là đồ thị sâu) thể hiện đồng thời nhiều mối quan hệ khác nhau giữa các sản phẩm thuộc nhiều phiên làm việc khác nhau trong bộ dữ liệu.

Đồ thị G

Gọi G là một đồ thị thoả mãn ma trận kề $M_G \in \mathbb{R}^{d\times d}$ với $M_G^{i,j}$ là số lần sản phẩm j được nhấp (click) ngay sau sản phẩm i trong một phiên. Khái niệm *ngay sau* thể hiện sự tương tác trực tiếp từ khi nhấp từ sản phẩm i tới sản phẩm j.

Giả sử $s = \{v_1, v_2, ..., v_n\}$ đại diện cho một phiên làm việc với v_i $(0 \le v_i < d)$ là đại diện cho một sản phẩn được nhấp thứ i. Lúc đó ta thực hiện, với mỗi i $(1 \le i < n)$:

$$M_G^{v_i,v_{i+1}} \leftarrow M_G^{v_i,v_{i+1}} + 1 \tag{1}$$

Nhận xét:

- G là đồ thị có hướng, có trọng số.
- Trong G, trọng số cạnh nối từ đỉnh i tới j có giá trị là $M_G^{i,j} \in \mathbb{R}$
- Theo thống kê, xác suất của để sản phẩm j được nhấp ngay sau sản phẩm i là:

$$P_G^{i,j} = \frac{M_G^{i,j}}{\sum_{x=0}^{d-1} M_G^{i,x}}$$

Đồ thị H

Gọi H là một đồ thị thoả mãn ma trận kề $M_H \in \mathbb{R}^{d\times d}$ với $M_H^{i,j}$ là số lần sản phẩm j được nhấp (click) sau khi nhấp sản phẩm i trong một phiên. Khái niệm sau khi thể hiện sự tương tác trực tiếp hoặc gián tiếp từ khi nhấp từ sản phẩm i tới sản phẩm j có thể đi qua một số sản phẩm nào khác giữa hai sản phẩm này.

Giả sử $s = \{v_1, v_2, ..., v_n\}$ đại diện cho một phiên làm việc với v_i $(0 \le v_i < d)$ là đại diện cho một sản phẩn được nhấp thứ i. Lúc đó ta thực hiện, với mỗi i, j $(1 \le i < j < n)$:

$$M_H^{v_i,v_j} \leftarrow M_H^{v_i,v_j} + 1 \tag{2}$$

Nhận xét:

- H là đồ thị có hướng, có trọng số.
- Trong H, trọng số cạnh nối từ đỉnh i tới j có giá trị là $M_H^{i,j} \in \mathbb{R}$
- $M_G^{i,j} \le M_H^{i,j} \quad \forall i, j : 0 \le i < j < d$
- Theo thống kê, xác suất của để sản phẩm j được nhấp sau sản phẩm i là:

$$P_H^{i,j} = \frac{M_H^{i,j}}{\sum_{x=0}^{d-1} M_H^{i,x}}$$

- Gọi E_X là số lượng cạnh của đồ thị X. Ta có, $E_H > E_G$.

Đồ thị K

Gọi c là số lượng nhấp nhiều nhất của một phiên trong cơ sở dữ liệu. Gọi K là một đồ thị thoả mãn ma trận kề $M_K \in \mathbb{R}^{d \times d \times c}$ với $M_K^{i,j}[t]$ là số lần sản phẩm j được nhấp (click) sau khi nhấp sản phẩm i đúng t lần nhấp trong một phiên.

Giả sử $s = \{v_1, v_2, ..., v_n\}$ đại diện cho một phiên làm việc với v_i $(0 \leq v_i < d)$ là đại diện cho một sản phẩn được nhấp thứ i. Lúc đó ta thực hiện, với mỗi i, j $(1 \leq i < j < n)$:

$$M_K^{v_i,v_j}[j-i-1] \leftarrow M_K^{v_i,v_j}[j-i-1] + 1 \tag{3}$$

Nhận xét:

- K là đồ thị có hướng, có trọng số.
- Trong K, trọng số cạnh nối từ đỉnh i tới j có giá trị là $M_K^{i,j} \in \mathbb{R}^c$. Vì thế, nó chiếm bộ nhớ lưu trữ, mất thời gian truy cập lấy giá trị hơn nhiều đồ thị H và G.
- Mang thông tin của cả 2 đồ thị H và G
- $M_G^{i,j} = M_K^{i,j}[0] \quad \forall i, j : 0 \leq i < j < d$
- $M_H^{i,j} = \sum_{t=0}^{c} M_K^{i,j}[t] \quad \forall i, j : 0 \leq i < j < d$
- Theo thống kê, xác suất của để sản phẩm j được nhấp sau khi nhấp sản phẩm i đúng t lần nhấp là:

$$P_K^{i,j}[t] = \frac{M_K^{i,j}[t]}{\sum_{x=0}^{d-1} M_K^{i,x}[t]}$$

- Gọi E_X là số lượng cạnh của đồ thị X. Ta có, $E_K = E_H > E_G$.

2. Mạng học sâu đồ thị (Graph Neural Network)

Mạng học sâu đồ thị (GNN) được giới thiệu đầu tiên vào năm 2005 [15], GNN là một loại mạng nơ-ron hoạt động trực tiếp trên cấu trúc đồ thị, với việc sử dụng nơ-ron như là các nút trong cấu trúc mạng, từng nút sẽ chứa thông tin của riêng nó và thu thập thêm các thông tin từ các nút lân cận thể hiện mối tương quan giữa các nút trong đồ thị.

Với hướng tiếp cận sử dụng đồ thị, GNN ngày càng trở nên phổ biến trong nhiều lĩnh vực khác nhau [16], tiềm năng của mô hình GNN cho thấy khả năng ứng dụng và xử lý được khá nhiều bài toán thực tế như xây dựng biểu đồ tri thức, đánh giá mối tương quan của mạng xã hội, hệ thống gợi ý bán hàng... Sức mạnh của GNN trong việc mô hình hóa được sự phụ thuộc giữa các nút trong đồ thị cho phép tạo ra bước đột phá trong lĩnh vực nghiên cứu liên quan đến phân tích đồ thị.

Trong nghiên cứu này, nhóm tác giả đề xuất sử dụng mô hình GNN để xây dựng mô hình gợi ý dựa và phiên làm việc của người dùng. Để có sự nhất quán trong phần mô tả các khái niệm, nhóm tác giả thiết lập một số thuật ngữ chính như sau: khái niệm đồ thị (*graph*) sẽ dùng để biểu diễn phiên làm việc (*session*), trong đó đỉnh (*node*) của đồ thị mô tả sản phẩm (*item*) lựa chọn và cạnh (*edge*) của đồ thị mô tả việc người dùng nhấp chuột (*click*) từ sản phẩm trước sang sản phẩm tiếp theo trong phiên.

Với phương án xây dựng mạng nơ-ron đồ thị GNN cho các đồ thị đơn quan hệ G, H và đồ thị sâu K mô tả ở phần trên, do K là đồ thị sâu nên cần có phương án phù hợp hơn để mô hình GNN có thể học được tính chất sâu của đồ thị K. Do đó, nhóm tác giả đề xuất hai mô hình khác nhau cho đồ thị đơn quan hệ và đồ thị sâu như sau:

Mô hình mạng nơ-ron cho đồ thị G và H

Mô hình mạng nơ-ron cho đồ thị đơn quan hệ G và H được biểu diễn như hình 2.

- c là số lượng nhấp được sử dụng làm đầu vào của mô hình, d là số lượng sản phẩm có trong toàn bộ phiên.
- Đầu vào của mô hình là một phiên gồm c nhấp có tính thứ tự lần là $s = \{id_1, id_2, ..., id_c\}$ với $0 \leq id_i \leq d$.
- Với mỗi nhấp id_i qua đồ thị, ta thu được một vector trọng số v_i với $v_i \in \mathbb{R}^d$.
- Sử dụng lớp *Norm* để chuẩn hóa v_i thành xác suất $p_i \in \mathbb{R}^d$ với công thức sau:

$$p_i = \frac{v_i}{sum(v_i)} \tag{4}$$

- Cuối cùng, sử dụng một lớp *Fully connected layer* với hàm kích hoạt là *softmax* để tính toán đầu ra của mô hình.

Mô hình mạng nơ-ron cho đồ thị K

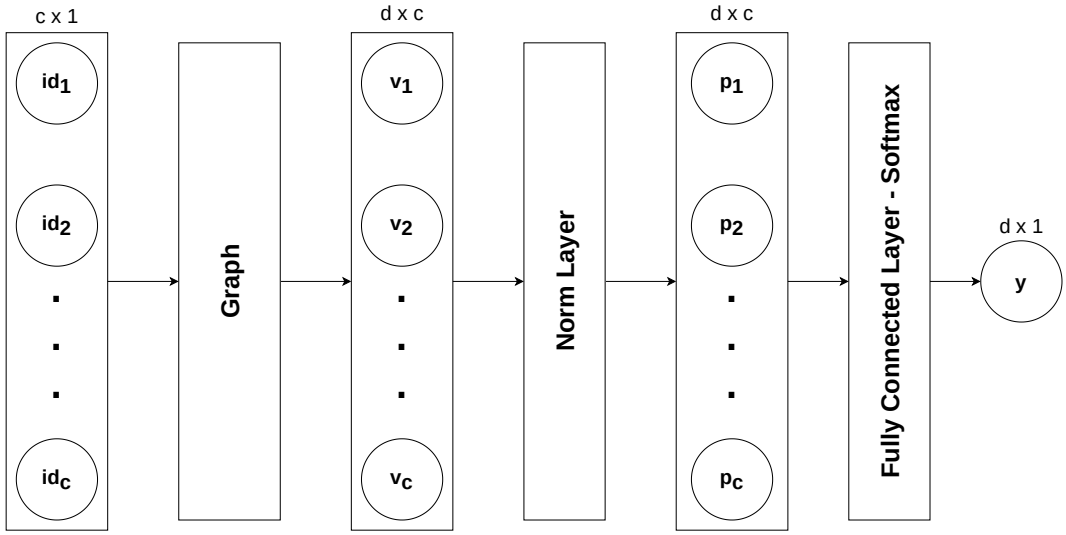
Để cải tiến mô hình mạng nơ-ron đồ thị khi phải làm việc với đồ thị K (tức đồ thị có chiều sâu), nhóm tác giả đề xuất sử dụng thêm một lớp học sâu (Depth layer) vào mô hình như hình 3.

- Đầu vào của mô hình là một phiên gồm c nhấp có tính thứ tự lần lượt là $s = \{id_1, id_2, ..., id_c\}$ với $0 \leq id_i \leq d$ (d là số lượng sản phẩm trong toàn bộ phiên).
- Với mỗi nhấp id_i qua đồ thị K, ta thu được một vector trọng số v_i với $v_i \in \mathbb{R}^{d \times c}$.
- Sử dụng lớp *Depth* để biến đổi chiều của v_i thành $\mathbb{R}^d$ với công thức sau:

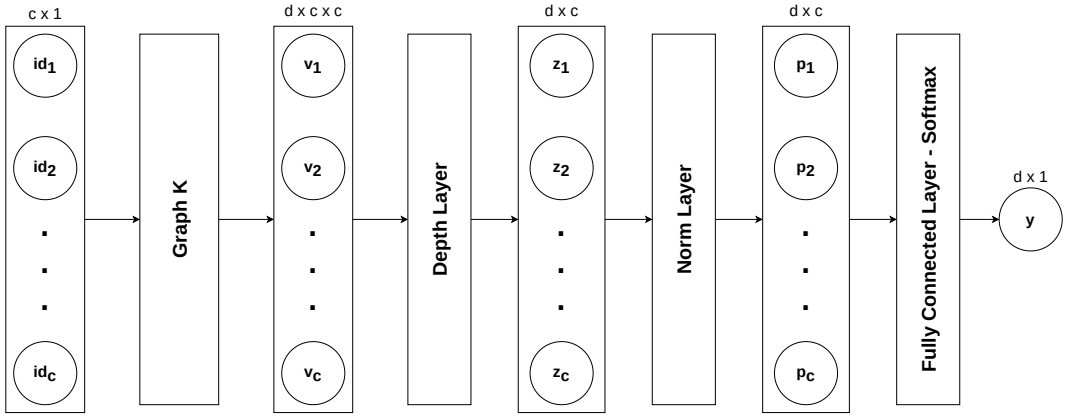
$$z_i = f(w_i v_i^T + b_i) \tag{5}$$

  Với:
  - $w_i \in \mathbb{R}^{1 \times c}$: trọng số chiều sâu
  - $b_i \in \mathbb{R}$: trọng số tự do của chiều sâu
  - $f(z)$: là một hàm biến đổi z, nhóm tác giả sử dụng hàm tuyến tính $f(z) = z$.
- Sử dụng lớp *Norm* để chuẩn hóa z_i thành xác suất $p_i \in \mathbb{R}^d$ với công thức sau:

$$p_i = \frac{z_i}{sum(z_i)} \tag{6}$$



Hình 2. Mô hình mạng nơ-ron cho đồ thị G và H



Hình 3. Mô hình mạng nơ ron cho đồ thị K

- Cuối cùng, sử dụng một lớp *Fully connected layer* với hàm kích hoạt là *softmax* để tính toán đầu ra của mô hình.

IV. NGUỒN DỮ LIỆU NGHIÊN CỨU

1. Nguồn Dữ Liệu (*Dataset*)

Nghiên cứu sử dụng bộ dữ liệu cung cấp bởi *YOOCHOOSE GmbH*, đây là bộ dữ liệu được sử dụng trong cuộc thi *RecSys Challenge 2015* [17]. Bộ dữ liệu ghi lại tập hợp nhiều phiên làm việc của một trang web thương mại điện tử hoạt động trong lĩnh vực bán lẻ tại châu Âu, trong đó mỗi phiên làm việc chứa thông tin về chuỗi nhấp chuột và một danh sách sản phẩn mà khách hàng lựa chọn trong suốt phiên đó. Dữ liệu được ghi nhận kéo dài trong 6 tháng, từ tháng 04/2014 đến tháng 09/2014. Vì lý do quyền riêng tư, toàn bộ thông tin về người sử dụng đã được ẩn đi khỏi bộ dữ liệu.

Bộ dữ liệu bao gồm hai tệp dữ liệu:

- **Dữ liệu nhấp chuột** (yoochoose-clicks.dat): chứa dữ liệu về chuỗi nhấp chuột của người dùng. Dữ liệu bao gồm các trường: (1) Session ID – ID của mỗi session. Trong mỗi session có thể có một hoặc nhiều sự kiện nhấp chuột. (2) Timestamp – thời gian xảy ra của sự kiện nhấp chuột. (3) Item ID – ID của sản phẩm được chọn. (4) Category – danh mục của sản phẩm được chọn.
- **Dữ liệu nhấp chuột đánh giá** (yoochoose-test.dat): giống với bộ dữ liệu nhấp chuột đã nêu ở trên nhưng những session trong đây không có trong tập trên. Nó được sử dụng để đánh giá mô hình.
- **Dữ liệu mua sắm** (yoochoose-buys.dat): chứa dữ liệu về chuỗi mua sắm của người dùng. Dữ liệu bao gồm các trường: (1) Session ID – ID của mỗi session. Trong mỗi session có thể có một hoặc nhiều sự kiện mua sắm.

(2) Timestamp – thời gian xảy ra của sự kiện mua sắm. (3) Item ID – ID của sản phẩm được mua. (4) Price – giá sản phẩm. (5) Quantity – số lượng sản phẩm được mua.

Mỗi Session ID trong yoochoose-buys.dat luôn xuất hiện trong yoochoose-clicks.dat – các dữ liệu cùng Session ID kết hợp lại tạo thành chuỗi nhấp chuột của một khách hàng cụ thể trong suốt một phiên làm việc. Thời gian của một phiên có thể rất ngắn (vài phút) hoặc rất dài (vài giờ), có thể bao gồm một hoặc nhiều sự kiện nhấp chuột và mua hàng, phụ thuộc vào hành vi tương tác của người sử dụng. Thông tin chi tiết về hai tập dữ liệu này được thể hiện ở Bảng I

Bảng I
THỐNG KÊ VỀ KÍCH THƯỚC BỘ DỮ LIỆU [17]

	yoochoose-clicks.dat	**yoochoose-buys.dat**
Số lượng sự kiện	33.003.944	1.150.753
Số lượng sản phẩm	52.739	19.949
Số lượng session	9.249.729	509.696

2. Một số phân tích bộ dữ liệu

Bảng thống kê dữ liệu của hai tập dữ liệu huấn luyện và kiểm thử được mô tả ở Bảng II.

Bảng II
THỐNG KÊ VỀ BỘ DỮ LIỆU NHẤP

	Bộ huấn luyện	**Bộ kiểm tra**	**Tất cả**
Số lượng phiên	9.249.729	2.312.432	11.562.161
Số lượng sản phẩm	52.739	42.155	54.287
Số lượng nhấp	33.003.944	8.251.791	41.255.735
Số nhấp lớn nhất	200	200	200
Số nhấp trung bình	3,5681	3,56845	3,56817
Số nhấp nhỏ nhất	1	1	1
Số phiên 1 nhấp	13,619%	13,602%	13,616%
Số phiên 2 nhấp	38,467%	38,463%	38,466%
Số phiên 3 nhấp	17,442%	17,467%	17,447%
Số phiên 4 nhấp	10,118%	10,099%	10,114%
Số phiên 5 nhấp	5,866%	5,903%	5,874%
Số phiên 6 nhấp	3,926%	3,914%	3,924%
Số phiên 7 nhấp	2,565%	2,566%	2,565%
Số phiên 8 nhấp	1,828%	1,833%	1,829%
Số phiên 9 nhấp	1,297%	1,298%	1,297%
Số phiên 10 nhấp	0,979%	0,962%	0,976%
Số phiên hơn 10 nhấp	3,892%	3,894%	3,892%

Với thống kê dữ liệu ở Bảng II, ta có một số nhận xét sau:

- Bộ dữ liệu chứa hơn 11 triệu phiên, trong đó bộ huấn luyện chứa hơn 9 triệu phiên và bộ kiểm tra chứa hơn 2 triệu phiên.
- Có tất cả 54.287 sản phẩm, bộ huấn luyện có 52.739 sản phẩm. Vì vậy có 1.548 sản phẩm có trong bộ kiểm tra mà không có trong bộ huấn luyện, dẫn đến việc không thể xác định (học) số sản phẩm này. Vì vậy, ta cần loại bỏ các phiên có sản phẩm này ra khỏi tập kiểm tra (các nghiên cứu liên quan về bộ dữ liệu này cũng loại bỏ dữ liệu này).
- Phiên có ít nhất 1 nhấp và có thể lên tới 200 nhấp.
- Số lượng phiên chỉ có 1 nhấp chiếm 13,6%, dữ liệu này gần như không có giá trị vì không đủ thông tin nên cần loại bỏ phiên này. Phiên làm việc có số lượng nhấp nhiều nhất là 2 nhấp và 3 nhấp với tỷ lệ lần lượt là 38,5% và 17,4%. Như vậy phiên có số lượng nhấp càng lớn thì càng ít xuất hiện.
- Trung bình mỗi phiên làm việc thì khoảng 4 nhấp (3,5)

3. Tiền xử lý dữ liệu

Các bước tiền xử lý được thực hiện như sau:

- Chuẩn hóa phiên:
  - Tổng hợp phiên theo danh sách nhấp (danh sách sản phẩm)
  - Loại bỏ một số thuộc tính dữ liệu không cần thiết như thời gian nhấp, catagory, ...
- Bỏ các phiên chỉ có 1 nhấp.
- Loại bỏ phiên làm việc trong bộ kiểm tra có chứa sản phẩm mà không xuất hiện trong bộ huấn luyện.

Bộ dữ liệu sau bước tiền xử lý được mô tả ở Bảng III.

4. Phương Án Chia Dữ Liệu

Bộ kiểm tra chứa tới gần 2 triệu phiên, đây là số lượng khá lớn nên phần thực nghiệm của nghiên cứu này chỉ lấy 25% của bộ kiểm tra làm tập kiểm tra cuối cùng (test) và 25% khác cũng thuộc bộ này làm tập đánh giá (validate). Lưu ý, tập kiểm tra và tập đánh giá có những phiên khác nhau và được chọn ngẫu nhiên. Ta có Bảng IV.

V. KẾT QUẢ THỰC NGHIỆM VÀ ĐÁNH GIÁ

1. Tham Số Đánh Giá Mô Hình

Trong các bài toán dự báo bán hàng thực tế, việc gợi ý cho khách hàng một sản phẩm tiếp theo thường không có lợi ích gì, thay vào đó hệ thống gợi ý cần đưa ra một danh sách k sản phẩm đề xuất tiếp theo. Vì lý do đó một số tham số cơ bản trong quá trình đánh giá mô hình như *Precision*, *Recall* hay *Accuracy* đơn lẻ không còn phù hợp nữa. Thay vào đó các nghiên cứu gần đề đây xuất sử

Bảng III
THỐNG KÊ VỀ BỘ DỮ LIỆU NHẤP SAU TIỀN XỬ LÝ

	Bộ huấn luyện	**Bộ kiểm tra**	**Tất cả**
Số lượng phiên	7.990.018	1.996.408	9.986.426
Số lượng sản phẩm	52.069	38.733	52.069
Số lượng nhấp	31.744.233	7.926.322	39.670.555
Số nhấp lớn nhất	200	200	200
Số nhấp trung bình	3,97299	3,97029	3,97245
Số nhấp nhỏ nhất	2	2	2
Số phiên 2 nhấp	44,532%	44,518%	44,529%
Số phiên 3 nhấp	20,192%	20,223%	20,198%
Số phiên 4 nhấp	11,713%	11,691%	11,709%
Số phiên 5 nhấp	6,791%	6,833%	6,800%
Số phiên 6 nhấp	4,545%	4,531%	4,543%
Số phiên 7 nhấp	2,969%	2,970%	2,969%
Số phiên 8 nhấp	2,117%	2,121%	2,117%
Số phiên 9 nhấp	1,502%	1,502%	1,502%
Số phiên 10 nhấp	1,133%	1,113%	1,129%
Số phiên hơn 10 nhấp	4,505%	4,498%	4,504%

Bảng IV
THỐNG KÊ VỀ BỘ KIỂM TRA VÀ ĐÁNH GIÁ

	Bộ kiểm tra	**Bộ đánh giá**
Số lượng phiên	499.102	499.102
Số lượng sản phẩm	30.179	30.278
Số lượng nhấp	1.982.109	1.978.213
Số nhấp lớn nhất	200	200
Số nhấp trung bình	3,971	3,963
Số nhấp nhỏ nhất	2	2
Số phiên 2 nhấp	44,580%	44,585%
Số phiên 3 nhấp	20,132%	20,146%
Số phiên 4 nhấp	11,643%	11,786%
Số phiên 5 nhấp	6,881%	6,769%
Số phiên 6 nhấp	4,532%	4,528%
Số phiên 7 nhấp	2,988%	2,979%
Số phiên 8 nhấp	2,129%	2,101%
Số phiên 9 nhấp	1,478%	1,519%
Số phiên 10 nhấp	1,133%	1,131%
Số phiên hơn 10 nhấp	4,504%	4,457%

dụng tham số $Recall@K$, $MRR@K$ và $ACCs@K$ để đánh giá việc gợi ý cùng lúc k sản phẩm. Trong nghiên cứu này, nhóm tác giả cũng đề xuất sử dụng các tham số này, trong đó:

Recall@K

Để đánh giá hiệu suất của mô hình với một hệ gợi ý, nghiên cứu sử dụng chỉ số $Recall@K$ với công thức sau:

$$Recall@K = \frac{1}{n}\sum_{i=0}^{n}\frac{|S^i_{rec} \cap S^i_{rel}|}{|S^i_{rel}|} \tag{7}$$

Trong đó, với n là số lượng phiên của bộ dữ liệu, S^i_{rec} là tập các sản phẩm gợi ý (gợi ý bởi $topK$ - K trọng số lớn nhất) và S^i_{rel} là tập các sản phẩm được nhấp thực tế của phiên thứ i với $0 \leq i < n$. Thông thường trong bài toán gợi ý, tập các nhấp thực tế (S^i_{rel}) chỉ có duy nhất một thành phần được gọi là next-click. Vì vậy, $S^i_{rel} = \{id_*\}$ với id_* là sản phẩm được nhấp gợi ý tiếp theo sau các sản phẩm đã nhấp trong phiên làm việc thứ i.

MRR@K

$MRR@K$ (Mean Reciprocal Rank) là mức trung bình của các cấp bậc tương hỗ của các sản phẩm mong muốn. Xếp hạng đối ứng được đặt thành 0 nếu thứ hạng lớn hơn K. MRR tính đến thứ hạng của mặt hàng, điều này rất quan trọng trong các cài đặt có thứ tự đề xuất quan trọng. $MRR@K$ có công thức tính như sau:

$$MRR@K = \frac{1}{n}\sum_{i=0}^{n} RR(id^i_*, S^i_{rec}) \tag{8}$$

Trong đó, với n là số lượng phiên của bộ dữ liệu, S^i_{rec} là tập K sản phẩm gợi ý được sắp xếp theo trọng số từ lớn đến bé (gợi ý bởi $top-K$ - trong đó K trọng số lớn nhất) và id_* là sản phẩm được nhấp gợi ý tiếp theo sau các sản phẩm đã nhấp trong phiên làm việc thứ i với $0 \leq i < n$. $RR(id, S)$ là 0 nếu sản phẩm id không có trong S, là $\frac{1}{r+1}$ với r là vị trí của id trong tập S tính từ 0.

Như vậy, nếu hệ gợi ý càng trả về đúng sản phẩm tiếp theo với điểm càng cao thì MRR càng cao. Lưu ý, chỉ số này chỉ áp dụng với một nhấp sản phẩm tiếp theo thực tế, không phù hợp cho việc gợi ý một chuỗi các nhấp (số nhấp lớn hơn 1).

ACCs@K

Bài báo này đề xuất một chỉ số $ACCs$ để tính độ chính xác trong một hệ gợi ý K nhãn với trọng số (xác suất) lớn nhất với nhiều nhãn thực tế. Đây là độ đo để tính cho một bài toán nhiều nhãn đầu ra (một quan sát nhưng có nhiều nhãn).

$$ACCs@K = \frac{1}{n}\sum_{i=0}^{n}\min(1, |S^i_{rec} \cap S^i_{rel}|) \tag{9}$$

Trong đó, với n là số lượng phiên của bộ dữ liệu, S^i_{rec} là tập các sản phẩm gợi ý (gợi ý bởi $topK$ - K trọng số lớn nhất) và S^i_{rel} là tập các sản phẩm được nhấp thực tế của

phiên thứ i với $0 \leq i < n$. Công thức $\min(1, |S^i_{rec} \cap S^i_{rel}|)$ chỉ ra rằng ở quan sát thứ i, có tồn tại 1 sản phẩm nào đó trong danh sách nhãn nằm trong K nhãn dự đoán có trọng số lớn nhất hay không, giá trị này bằng 1 nếu có tồn tại và ngược lại bằng 0.

2. Kết quả và Nhận xét

Trong thí nghiệm này nhóm tác giả có chạy thử nghiệm với mô hình thống kê để làm kết quả cơ sở của thí nghiệm. Kết quả của quá trình huấn luyện của hai mô hình với ba đồ thị G, H, K được mô tả ở Bảng V:

Bảng V
BẢNG KẾT QUẢ TỔNG QUAN

Mô hình	Stats.H	Stats.G	GNN.G	GNN.H	GNN.K
Recall@1	0,227	0,223	0,218	0,214	0,221
Recall@5	0,500	0,505	0,518	0,512	0,519
Recall@10	0,610	0,615	0,625	0,620	0,626
Recall@20	0,696	0,701	0,708	0,703	0,709
ACCs@1	0,253	0,248	0,241	0,239	0,245
ACCs@5	0,533	0,538	0,549	0,545	0,550
ACCs@10	0,639	0,643	0,651	0,647	0,652
ACCs@20	0,719	0,723	0,728	0,724	0,729
MRR@1	0,227	0,223	0,218	0,214	0,221
MRR@5	0,327	0,327	0,330	0,325	0,332
MRR@10	0,342	0,342	0,344	0,339	0,346
MRR@20	0,348	0,348	0,350	0,345	0,352

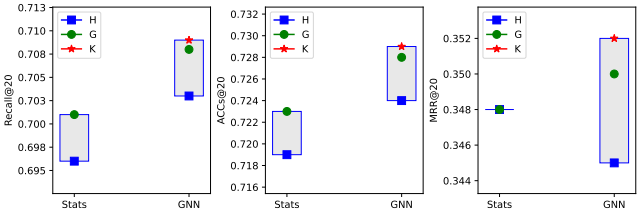
Qua Bảng V ta thấy:

- Thử nghiệm với 5 mô hình khác nhau, gồm 2 mô hình thống kê truyền thống và 3 mô hình học sâu sử dụng đồ thị. Lưu ý mô hình thống kê không hỗ trợ làm việc với đồ thị sâu K.
- Đánh giá mô hình dựa trên 3 tham số chính là $Recall@K$, $ACCs@K$, $MRR@K$ với $K \in [1, 5, 10, 20]$
- Mô hình thống kê với đồ thị H (*Stats.H*) có kết quả tệ nhất, còn tốt nhất là mô hình sử dụng mạng học sâu GNN với đồ thị K (*GNN.K*).
- Tuy nhiên kết quả cho thấy khoảng cách giá trị giữa các mô hình với từng tham số không lớn.

Hình 4 biểu diễn trực quan kết quả so sánh của từng nhóm mô hình khi thực hiện với các loại đồ thị khác nhau, trong khuôn khổ bài báo này tác giả chỉ biểu diễn với top-K=20, các kết quả của K còn lại có thể tham khảo ở Bảng V:

3. So sánh với các nghiên cứu liên quan

Để so sánh hướng tiếp cận và kết quả đạt được của nghiên cứu này, tác giả lựa chọn một số bài báo tương tự cùng giải



Hình 4. Biểu đồ kết quả cho các mô hình với top-K=20

quyết bài toán hệ gợi ý dựa vào phiên làm việc và cùng sử dụng bộ dữ liệu *Yoochoose* của cuộc thi RecSys Challenge 2015 [17] như trong nghiên cứu này.

Hai nghiên cứu của Balázs Hidasi (2015) [8] và Yong Kiam Tan (2016) [9] đều sử dụng mạng nơ ron hồi quy (RNN) cho bài toán này, trong đó Yong Kiam Tan đã đề xuất cải tiến mô hình RNN với một thuật toán làm giàu dữ liệu và cho kết quả tốt hơn hẳn so với mô hình RNN của Balázs Hidasi. Thuật toán làm giàu dữ liệu trong quá trình tiền xử lý dữ liệu của Y.K. Tan và cộng sự để sinh dữ liệu là: với một phiên $s = [v_{s,1}, v_{s,2}, ..., v_{s,n}]$ thì sẽ tạo ra một chuỗi phiên con và nhãn là $([v_{s,1}], v_{s,2}), ([v_{s,1}, v_{s,2}], v_{s,3}), ..., ([v_{s,1}, v_{s,2}, ..., v_{s,n-1}], v_{s,n})$ với $[v_{s,1}, v_{s,2}, ..., v_{s,n-1}]$ là chuỗi nhấp đầu vào và $v_{s,n}$ là nhãn $next - click$. Điểm lưu ý là bộ dữ liệu này có số sản phẩm là 37.483 sau quá trình tiền xử lý, khác với thống kê gốc của bộ dữ liệu này là 52.739 sản phẩm [17], điểm khác biệt này sẽ là đáng kể với các mô hình gợi ý phân lớp đa nhãn.

Do bộ dữ liệu này khá lớn, Y.K. Tan và cộng sự gặp một số khó khăn trong quá trình huấn luyện khi bị giới hạn phần cứng, có lẽ vì lý do đó Y.K. Tan và cộng sự đưa ra ý tưởng chia nhỏ bộ dữ liệu để thực nghiệm thành các bộ $Yoochoose$ nhỏ hơn (1/4, 1/16, 1/64, 1/256). Trong quá trình thực nghiệm, Y.K. Tan và cộng sự nhận thấy việc sử dụng bộ dữ liệu đầy đủ mang đến kết quả kém hơn so với việc dùng một phần của dữ liệu. Lý do chính mà Y.K. Tan và cộng sự đưa ra nhận xét này là do số lượng nhãn quá lớn trên bộ dữ liệu đầy đủ, con số này sẽ bị giảm đáng kể khi tách thành từng phần nhỏ hơn nên mô hình sẽ học nhẹ nhàng hơn nhiều. Cho dù Y.K. Tan và cộng sự kết luận mô hình $M2$ cho kết quả tốt nhất với bộ dữ liệu con $Yoochoose1/64$ với $Recall@20$ là $0,7129$ và $MRR@20$ là $0,3091$, tuy nhiên mô hình này không thể thực hiện được trên bộ dữ liệu đầy đủ do giới hạn phần cứng. Còn với bộ kết dữ liệu đầy đủ, mô hình M3 của Y.K. Tan và cộng sự cho kết quả tốt nhất là $Recall@20$ xấp xỉ $0,680$ và $MRR@20$ xấp xỉ $0,290$.

Jing Li và các cộng sự (2017) [11] đề xuất sử dụng mô hình NARM (*Neural Attentive Recommendation Machine*) để xây dựng hệ gợi ý phiên làm việc và cũng sử dụng bộ dữ

liệu con *Yooochoose* 1/4 và 1/64. Thực nghiệm của Jing Li sử dụng bộ dữ liệu kiểm tra với 55.898 phiên, số lượng sản phẩm là 16.766 với thực nghiệm 1/64 và 29.618 với thực nghiệm 1/4. Jing Li đạt được kết quả $Recall@20$ là 0,6973 và $MRR@20$ là 0,2923, với bộ dữ liệu 1/4. Trong bảng so sánh kết quả với nghiên cứu sử dụng RNN của Hidasi [8] và Y.K. Tan và cộng sự [9], Jing Li cho rằng kết của mình tốt hơn, dù rằng nếu so sánh kỹ thì kết của của Jing Li không thực sự đầy đủ và nổi trội như nghiên cứu của Tan, ví dụ như không thực nghiệm với bộ dữ liệu đầy đủ.

Cũng với hướng tiếp cận sử dụng bộ dữ liệu con của Tan, Shu Wu và cộng sự (2019) [12] sử dụng mô hình mạng học sâu đồ thị (GNN) cùng khá nhiều kỹ thuật xử lý đồ thị và các biến thể khác nhau của GNN để phân tích bài toán gợi ý dựa vào phiên làm việc. Nhóm nghiên cứu này đã đề xuất một mô hình sử dụng kỹ thuật biến đổi véc tơ phiên làm việc sang một không gian nhúng bằng cách sử dụng mạng đồ thị để huấn luyện và học. Khác với các nghiên cứu trước sử dụng tham số $Recall@K$, Shu Wu đề xuất sử dụng tham số $Precison@K$ và một điểm lưu ý là Shu Wu lại so sánh kết quả $Precision@K$ của mình với $Recall@K$ của các nghiên cứu trước đây, nên khả năng có sự nhầm lẫn trong nghiên cứu này và việc so sánh kết quả là không hợp lý.

Trong nghiên cứu của Kiewan và cộng sự (2018) [10] cũng sử dụng mô hình RNN như hai bài báo của Balázs Hidasi [8] và Yong Kiam Tan [9], tuy nhiên Kiewan không chia nhỏ bộ dữ liệu như Y.K. Tan và cộng sự mà có hướng tiếp cận khác biệt là đưa ra khái niệm phiên dài và phiên ngắn (*long and short sequential session*), cụ thể Kiewan đánh giá mô hình theo 3 nhóm phiên có độ dài là [2-5], [6-25] và [26-200]. Kiewan đề xuất một số kỹ thuật khác nhau (bao gồm chuẩn hóa lớp LN, ma trận nhúng đầu vào RE hoặc xếp chồng lớp GRU) để xây dựng mô hình phù hợp với các loại dữ liệu phiên có độ dài ngắn khác nhau. Mô hình của Kiewan cho kết quả tốt nhất $Recall@20$ là 0,691. Như vậy, kết quả này là tốt hơn với kết quả của Y.K. Tan và cộng sự khi kiểm tra với bộ dữ liệu đầy đủ nhãn có $Recall@20$ xấp xỉ 0,680 và tốt hơn hẳn so với kết quả của Balázs Hidas với $Recall@20$ là 0,632. Nếu so sánh kết quả này với kết quả tốt nhất của Y.K. Tan và cộng sự với tập dữ liệu con 1/64 với số nhãn ít hơn thì không tốt bằng ($Recall@20$ trên tập 1/64 là 0,7129), tuy nhiên việc so sánh như vậy cũng không hoàn toàn hợp lý do số lượng nhãn của tập dữ liệu đầy đủ mà Kiewan kiểm tra (khoảng 37 nghìn nhãn) nhiều hơn so với tập con 1/64 của Y.K. Tan và cộng sự (gần 17 nghìn nhãn).

Sau khi phân tích các nghiên cứu liên quan ở trên, tác giả có một số nhận xét như sau về kết quả của mình so với các nghiên cứu trước đây:

- Nghiên cứu này sử dụng cả tập dữ liệu huấn luyện và kiểm thử từ bộ dữ liệu gốc với số lượng sản phẩm, tức số lượng nhãn, lên tới hơn 52 nghìn.
  Các nghiên cứu trước đây không sử dụng bộ dữ liệu kiểm thử riêng biệt của bộ dữ liệu gốc, mà trích ra từ tập dữ liệu huấn luyện. Điều này làm giảm số lượng sản phẩm, tức số lượng nhãn của mô hình xuống còn từ 10 tới 37 nghìn nhãn.
- Nghiên cứu này đề xuất và xây dựng mô hình có tính mở rộng cao khi hoạt động với đồ thị với hơn 52 nghìn đỉnh, đặc biệt với số lượng cạnh rất lớn với các loại đồ thị gián tiếp H hay đồ thị sâu K.
  Một số nghiên cứu liên quan trình bày không thể chạy được mô hình với bộ dữ liệu đầy đủ, do đó họ phải thực nghiệm với bộ dữ liệu nhỏ hơn với số lượng nhãn thậm chí còn ít hơn.
- Mô hình đề xuất của nghiên cứu này cho kết quả $Recall@20$ là $0,712$ và $MRR@20$ là $0,363$, tốt hơn kết quả của Kiewan có $Recall@20$ là $0,691$ và của Y.K. Tan và cộng sự có $Recall@20$ là $0,680$ (Y.K. Tan và cộng sự có nhiều kết quả khác nhau, đây là kết quả chạy với bộ dữ liệu đầy đủ), và tốt hơn hẳn nghiên cứu đầu tiên của Balázs Hidas với $Recall@20$ là $0,632$.

VI. KẾT LUẬN

Nghiên cứu này có hai đóng góp chính gồm: (1) sử dụng đồ thị để mô tả chuỗi nhấp chuột trong quá trình lựa chọn sản phẩm trong phiên làm việc hiện tại, bao gồm ba loại đồ thị trực tiếp G, gián tiếp H và sâu K và (2) sử dụng mạng học sâu đồ thị GNN để học dữ liệu đồ thị để đưa ra các mô hình gợi ý dựa theo bài toán $top-K$. Thực nghiệm cho thấy hướng đề xuất của nghiên cứu này cho kết quả $Recall@20$ và $MRR@20$ tốt hơn hẳn các nghiên cứu trước đây, đặc biệt là khả năng xử lý đồ thị với số lượng đỉnh rất lớn và có khả năng mở rộng hơn nữa, đây chính là điểm hạn chế của các nghiên cứu trước đây gặp nhiều khó khăn khi phải xử lý bộ dữ liệu có số lượng sản phẩm (nhãn) khá lớn với hơn 50 nghìn nhãn, một số nghiên cứu trước đây còn phải chia nhỏ bộ dữ liệu và như vậy đã phải giảm nhãn của mô hình gợi ý.

Hướng mở rộng của nghiên cứu này là tiếp tục sử dụng một số kỹ thuật biến đổi đồ thị (ví dụ phép nhúng đồ thị) sao cho mô hình GNN hoạt động hiệu quả hơn, hoặc phương án thiết kế tối ưu lớp sâu của mô hình GNN để nó có thể hoạt động hiệu quả hơn với đồ thị sâu K.

TÀI LIỆU THAM KHẢO

[1] J. B. Schafer, J. Konstan, and J. Riedl, "Recommender systems in e-commerce," in *Proceedings of the 1st ACM conference on Electronic commerce*, 1999, pp. 158–166.

[2] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[3] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[4] J. B. Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation applications," *Data mining and knowledge discovery*, vol. 5, no. 1, pp. 115–153, 2001.

[5] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender system-a case study," Minnesota Univ Minneapolis Dept of Computer Science, Tech. Rep., 2000.

[6] B. M. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering," in *Proceedings of the fifth international conference on computer and information technology*, vol. 1. Citeseer, 2002, pp. 291–324.

[7] Z. Huang, W. Chung, and H. Chen, "A graph model for e-commerce recommender systems," *Journal of the American Society for information science and technology*, vol. 55, no. 3, pp. 259–274, 2004.

[8] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," 2015. [Online]. Available: https://arxiv.org/abs/1511.06939

[9] Y. K. Tan, X. Xu, and Y. Liu, "Improved recurrent neural networks for session-based recommendations," *CoRR*, vol. abs/1606.08117, 2016. [Online]. Available: http://arxiv.org/abs/1606.08117

[10] K. Villatel, E. Smirnova, J. Mary, and P. Preux, "Recurrent neural networks for long and short-term sequential recommendation," *CoRR*, vol. abs/1807.09142, 2018. [Online]. Available: http://arxiv.org/abs/1807.09142

[11] J. Li, P. Ren, Z. Chen, Z. Ren, and J. Ma, "Neural attentive session-based recommendation," *CoRR*, vol. abs/1711.04725, 2017. [Online]. Available: http://arxiv.org/abs/1711.04725

[12] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 346–353.

[13] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, R. Anil, Z. Haque, L. Hong, V. Jain, X. Liu, and H. Shah, "Wide & deep learning for recommender systems," 2016.

[14] K. Nguyen, A. Nguyen, L. Vu, N. Mai, and B. Nguyen, "An efficient deep learning method for customer behaviour prediction using mouse click events," 11 2018.

[15] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 2, pp. 729–734 vol. 2, 2005.

[16] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE transactions on neural networks*, vol. 20, no. 1, pp. 61–80, 2008.

[17] Yoochoose Dataset, "Recsys challenge," 2015, https://github.com/RUCAIBox/RecSysDatasets/tree/master/dataset_info/YOOCHOOSE.

SƠ LƯỢC VỀ TÁC GIẢ



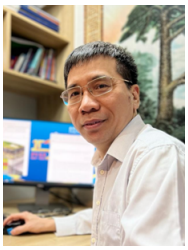
Nguyễn Tuấn Khang là nghiên cứu sinh tiến sỹ tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học và công nghệ Việt Nam. Hiện đang là tư vấn giải pháp công nghệ tại IBM Việt Nam. Lĩnh vực nghiên cứu: tư vấn các giải pháp công nghệ thông tin trong lĩnh vực tài chính ngân hàng, chuyên sâu vào mảng học sâu và phân tích hành vi.
Email: khang_nt@yahoo.com



Nguyễn Tú Anh tốt nghiệp trường Đại học Thăng Long năm 2020, hiện đang theo học chương trình thạc sỹ tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học và công nghệ Việt Nam. Hiện đang là giảng viên khoa Công nghệ thông tin của trường Đại học Thăng Long. Lĩnh vực nghiên cứu: thị giác máy tính và phân tích dữ liệu.
Email: anh.nt@thanglong.edu.vn



Mai Thúy Nga đạt học vị Tiến sỹ ngành Công nghệ thông tin tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học và công nghệ Việt Nam năm 2017. Hiện đang là giảng viên khoa Công nghệ thông tin của trường Đại học Thăng Long. Lĩnh vực nghiên cứu: xây dựng chương trình đào tạo đại học, phân tích thiết kế phần mềm, hệ thống thông tin quản lý, trí tuệ nhân tạo và khoa học dữ liệu.
Email: ngamt@thanglong.edu.vn



Nguyễn Hải An đạt học vị Tiến sỹ ngành Kỹ thuật khai thác dầu khí tại trường ĐH Mỏ Địa chất Hà Nội năm 2012; thạc sỹ Công nghệ dầu khí và phát triển mỏ tại Viện dầu lửa Pháp. Hiện là trưởng phòng Khoa học công nghệ, TCT Thăm dò Khai thác Dầu khí. Lĩnh vực nghiên cứu: ứng dụng kỹ thuật mới, công nghệ thông minh trong tối ưu phát triển khai thác các mỏ trên thềm lục địa Việt Nam và nâng cao hệ số thu hồi dầu và khí.
Email: annh1@pvep.com.vn



Nguyễn Việt Anh đạt học vị Tiến sỹ ngành Khoa học máy tính tại đại học Kyoto, Nhật bản và được phong hàm Phó giáo sư năm 2022 tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học và công nghệ Việt Nam. Hiện là nghiên cứu viên cao cấp tại Viện Công nghệ thông tin - Viện Hàn lâm khoa học và công nghệ Việt Nam. Lĩnh vực nghiên cứu: học máy, dữ liệu lớn và phân tích mạng xã hội.
Email: anhnv@ioit.ac.vn

Một phương pháp xây dựng hệ thống nhận dạng hành vi của người trong thời gian thực dựa trên các thuật toán học máy

Nguyễn Quang Huy[1], Trần Đức Nghĩa[1,*], Đào Tô Hiệu[2], Trần Đức Tân[2], Vũ Thị Thương[3], Đỗ Xuân Trung[4]

[1] Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội

[2] Trường Đại học Phenikaa, Hà Nội

[3] Đại học Phương Đông, Hà Nội

[4] Đại học Kinh Bắc, Bắc Ninh

Tác giả liên hệ: Trần Đức Nghĩa, Email: nghiatd@ioit.ac.vn

Ngày nhận bài: 31/08/2022, ngày sửa chữa: 19/11/2022, ngày duyệt đăng: 25/11/2022

Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1137

Tóm tắt: Một trong những vấn đề quan trọng trong hệ thống nhận dạng hành vi của người (Human Áctivity Recognition – HAR) đó là giá thành, tốc độ và độ chính xác. Cùng với sự phát triển của các thiết bị di động, các bộ vi xử lý ngày càng hiện đại đi cùng với việc tích hợp nhiều cảm biến khác nhau bao gồm cả gia tốc kế đã giúp việc nhận dạng hành vi người trở nên dễ dàng và nhanh chóng hơn. Tuy nhiên, độ chính xác của việc phân loại hành vi dựa trên cảm biến gia tốc còn phụ thuộc rất nhiều yếu tố khác nhau dẫn đến kết quả có thể có sự sai lệch. Bằng nhiều thử nghiệm khác nhau, chúng tôi nhận thấy rằng việc lựa chọn các thuộc tính tối ưu có quan hệ quyết định tới hiệu suất của việc phân loại. Trong bài báo này, chúng tôi đề xuất một bộ thuộc tính đơn giản nhưng cho kết quả rất khả quan với độ chính xác tới 99%. Chúng tôi đã thực nghiệm và so sánh với một vài thuật toán phân loại khác nhau trên tập dữ liệu thu thập từ một số tình nguyện viên.

Từ khóa: *Phân loại, cảm biến gia tốc, nhận dạng hành vi.*

Title: **A Method of Building a Real-time Human Behavior Recognition System based on Machine Learning Algorithms**

Abstract: One of the key issues in Human Activity Recognition (HAR) systems is cost, speed, and accuracy. Along with the development of mobile devices, integration of various sensors, including accelerometers, has made human behavior recognition easier and faster. However, the accuracy of the behavior classification based on the accelerometer depends on many factors, leading to misleading results. In various tests, we found that the selection of optimal attributes has a decisive relationship to performing the classifier. In this paper, we propose a simple set of attributes but give very satisfactory results with 99% accuracy. We have tested and compared with a few different classification algorithms on a dataset collected from several volunteers.

Keywords: *Classification, acceleration sensor, activity recognition*

I. GIỚI THIỆU

Bài toán nhận dạng hoạt động người nói chung và nhận dạng hoạt động dựa trên cảm biến gia tốc nói riêng đang chứng tỏ được tầm quan trọng của nó với nhiều ứng dụng trong các lĩnh vực như nông nghiệp, y tế, công nghiệp, an ninh. [1-3]. Tuy nhiên, dù ở lĩnh vực nào thì việc thu thập dữ liệu cũng là yếu tố hết sức quan trọng. Ngày nay các thiết bị cảm biến được tích hợp nhiều trong các thiết bị di động để theo dõi các chỉ số khác nhau. Chẳng hạn như điện thoại thông minh không chỉ là thiết bị giao tiếp thông thường mà còn có thể thu nhận nhiều tín hiệu môi trường từ các cảm biến bên trong nó [4]. Một trong số đó, gia tốc kế là cảm biến được sử dụng thường xuyên nhất cho việc nhận dạng hoạt động vì khả năng mang lại độ chính xác cao với năng lượng sử dụng thấp [5]. Với ưu điểm như vậy, gia tốc kế đã được nghiên cứu để thu thập dữ liệu từ nhiều vị trí khác nhau như chân [6], cổ tay [7], cánh tay [8], thắt lưng [9]. Ngoài ra, nhiều nghiên cứu đã sử dụng kết hợp cùng lúc nhiều cảm biến khác nhau để có thể phân loại được nhiều hành động hơn. Muhammad Shoaib cùng cộng sự [10] đã chứng minh rằng việc kết hợp con quay

hồi chuyển, từ kế và gia tốc kế có tác dụng tương hỗ làm cải thiện khả năng nhận dạng. Lei Gao và nhóm của mình trong công bố [11] đã nghiên cứu phân loại 8 hành động với bốn cảm biến gia tốc đặt ở bốn vị trí khác nhau là ngực, dưới cánh tay trái, eo và đùi. Mặc dù phương pháp đa cảm biến này có ưu điểm là gia tăng độ chính xác và khả năng nhận dạng được nhiều hành vi khác nhau, nhưng chúng vẫn có nhược điểm như giá thành thiết bị cao, tiêu hao nhiều năng lượng và hơn cả là gây khó chịu cho người mang nó, nhất là trong thời gian theo dõi dài. Trong phạm vi bài báo này, chúng tôi sử dụng mô hình nhận dạng dựa trên hành vi. So với các mô hình khác như mô hình hóa dựa trên vector (hoạt động, vị trí), mô hình hóa dựa trên vị trí, mô hình nhận dạng dựa trên hành vi đã được chứng minh đạt được độ chính xác cao nhất và ít tốn thời gian nhất, có thể xác định hiệu quả hành vi của con người [12]. Ngoài ra, các hành động cũng được giới hạn trong đời sống hàng ngày như đi lại, đứng, ngồi, nằm. Việc giới hạn này nhằm đưa ra một giải pháp xử lý tối ưu cho các thiết bị phần cứng vốn giới hạn ở khả năng xử lý và tính di động mà chúng tôi có. Do đó, bài báo này đề xuất sử dụng cảm biến gia tốc trên điện thoại thông minh đeo ở thắt lưng, thực hiện các thuật toán nhẹ về tính toán để nhận dạng các hoạt động cơ bản hằng ngày của con người. Phương pháp được đề xuất sử dụng một tập các thuộc tính tối thiểu để hệ thống có thể nhận dạng trong thời gian thực nhưng vẫn cho kết quả nhận dạng tốt.

II. PHƯƠNG PHÁP NGHIÊN CỨU

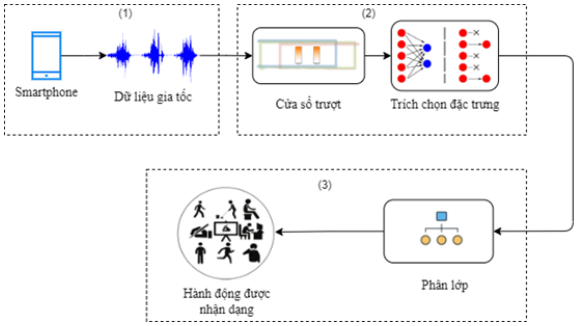
1. Mô hình nhận dạng

Kiến trúc của mô hình chúng tôi sử dụng để nhận dạng hoạt động người được minh họa trong Hình 1. Nó bao gồm ba thành phần chính:

(1) Thu thập dữ liệu: thu thập các tín hiệu thô từ cảm biến gia tốc trên điện thoại thông minh của người dùng (yêu cầu điện thoại có cảm biến gia tốc ba chiều và đã được cài đặt ứng dụng thu thập dữ liệu). Dữ liệu thu được là dữ liệu gia tốc theo ba trục X, Y, X được tiền xử lý thành luồng dữ liệu.

(2) Trích chọn các đặc trưng: Luồng dữ liệu ở trên được phân tách bằng cửa sổ trượt để lấy một số dữ liệu chuỗi trong thời gian ngắn. Sau đó dữ liệu này được đưa vào bộ trích chọn các đặc trưng và chọn lọc để đưa vào bộ phân loại.

(3) Phân loại hành động: Trong phần này, bộ phân loại được huấn luyện để xây dựng mô hình phân loại hành vi dựa trên các thuật toán học máy.



Hình 1. Mô hình hệ thống nhận dạng hành vi người

2. Thu thập dữ liệu

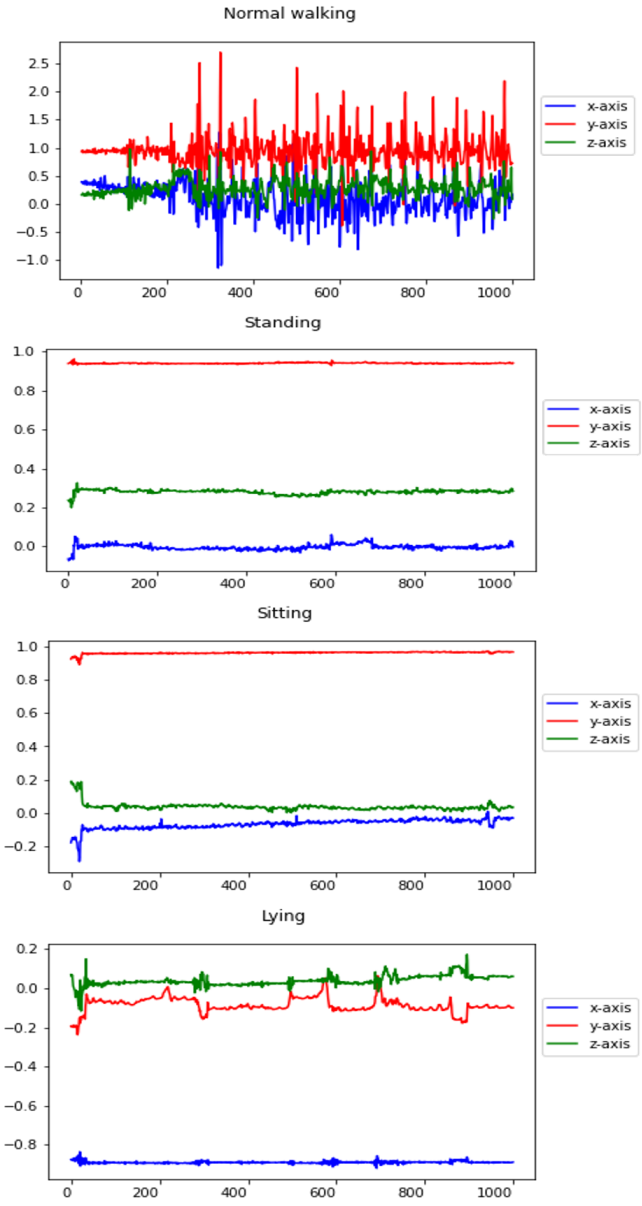
Để thu thập bộ dữ liệu hoạt động, chúng tôi sử dụng điện thoại thông minh Android (Note10 Lite) có cài đặt ứng dụng Sensor Data Collector ghi lại dữ liệu từ gia tốc kế. Ba tình nguyện viên tham gia nghiên cứu này là người lớn có độ tuổi, chiều cao và cân nặng khác nhau. Mỗi người thực hiện các hành vi cơ bản trong cuộc sống hàng ngày là đi, đứng, nằm, ngồi một cách tự nhiên. Trong khi hoạt động, điện thoại được gắn cố định trên thắt lưng của mỗi người. Quá trình thu thập dữ liệu, nếu điện thoại ở trạng thái sử dụng màn hình hoặc nhận cuộc gọi, ứng dụng sẽ tạm ngừng việc thu thập. Số lượng dữ liệu thu được trong một giây phụ thuộc vào tần số lấy mẫu của thiết bị. Chúng tôi cài đặt tần số 1Hz cho cảm biến gia tốc để ghi lại dữ liệu hoạt động. Một số nghiên cứu gần đây cũng chứng minh rằng các nhiệm vụ phân loại đơn giản có thể được tiến hành hiệu quả ở tần số lấy mẫu và độ phân giải thấp, làm tăng tuổi thọ pin của thiết bị [13, 14]. Bộ dữ liệu thực nghiệm này được thể hiện ở Bảng I.

Bảng I
SỐ LƯỢNG THỬ NGHIỆM

Hoạt động	**Số lượng phục vụ huấn luyện mô hình**	**Số lượng kiểm thử**	**Tổng**
Nằm	11035	7430	15765
Ngồi	7178	3076	10254
Đứng	6709	2875	9584
Đi bộ	8253	3537	11790
Tổng	**33175**	**14218**	**47393**

Hình 2 thể hiện dữ liệu thô từ các cảm biến tương ứng với mỗi hoạt động khác nhau. Đối với các hoạt động tĩnh (nằm, ngồi, đứng), chúng tôi nhận thấy rằng không có sự giao nhau giữa các trục, do các cảm biến có xu hướng ổn định hơn các hoạt động khác. Trong khi đó, với các hoạt động di chuyển, dễ dàng nhận thấy sự chồng chéo các chỉ

số của cảm biến. Điều đó là bình thường và chứng tỏ là các cảm biến đang hoạt động tốt.



Hình 2. Dữ liệu gia tốc của các hoạt động: (Lying) – Nằm, (Standing) – Đứng, (Sitting) – Ngồi, (Normal Walking) – Di chuyển

3. Phân đoạn dữ liệu

Dữ liệu thu được từ cảm biến gia tốc là các dữ liệu thô có bản chất là các dao động theo thời gian. Các mô hình phân loại không thể được áp dụng trực tiếp trên các dữ liệu thô này. Nguyên nhân là do một hoạt động có thể kéo dài vài giây hoặc hơn nữa, nên một điểm dữ liệu riêng lẻ không thể chứa đầy đủ thông tin mô tả cho hoạt động đã thực hiện. Phân đoạn là quá trình nhóm dữ liệu cảm biến thô của chuỗi thời gian liên tục thành các phần có thể quản lý được chứa thông tin mô tả một hoạt động [15]. Trong đó, phương pháp phân đoạn bằng cửa sổ thời gian đã được chứng minh là cho kết quả tốt nhất với việc nhận dạng các hoạt động thường ngày của con người [16].

Số lượng dữ liệu mẫu trong một phân đoạn hoặc cửa sổ phụ thuộc vào độ dài cửa sổ được xác định trước. Việc lựa chọn tham số cửa sổ thời gian phù hợp cũng có hai cách tiếp cận. Cách thứ nhất là thu thập luồng dữ liệu theo kích thước cửa sổ thời gian cố định. Cách thứ hai là cửa sổ thời gian chồng chéo (hay cửa sổ trượt), tức là dữ liệu cửa sổ trước đó được chồng lên dữ liệu cửa sổ hiện tại theo tỷ lệ phần trăm được xác định trước [15]. Chúng tôi lựa chọn sử dụng phương pháp cửa sổ trượt bởi đây là phương pháp phù hợp cho việc nhận dạng các hành động động và tĩnh trong thời gian thực [17]. Để lựa chọn độ dài cửa sổ tối ưu, chúng tôi đã tiến hành thử nghiệm với nhiều kích thước khác nhau (1s – 10s) với các khoảng chồng chéo (0% - 95%). Chúng tôi nhận thấy nếu độ dài cửa sổ quá ngắn thì sẽ không đủ thông tin để mô tả các hoạt động. Ngược lại, nếu kích thước của sổ quá lớn, dữ liệu có thể bị nhiễu bởi trong thời gian đó đối tượng có thể đã thay đổi các hoạt động khác nhau. Như vậy, không có độ dài cửa sổ nào là thích hợp cho mọi tình huống hoạt động của con người. Tuy nhiên, trong thực nghiệm mà chúng tôi tiến hành là nhận dạng các hoạt động cơ bản hàng ngày với thời gian thực, chúng tôi lựa chọn độ dài cửa sổ là 3s với khoảng chồng chéo là 50%.

4. Trích chọn đặc trưng

Dữ liệu thô từ cảm biến có thể phân tích thành rất nhiều các đặc trưng khác nhau. Trích chọn đặc trưng là một kỹ thuật trong miền chuyên biệt (như thời gian hoặc tần số) định nghĩa một đặc trưng mới từ dữ liệu thô nhằm giảm độ phức tạp tính toán và nâng cao quy trình nhận dạng [18, p. 193]. Trước đây, nhiều kỹ thuật trích chọn đặc trưng phức tạp như phân tích thành phần chính (PCA) [19], phân tích phân biệt tuyến tính (LDA) [20] và các đặc trưng wavelet [21] đã được sử dụng; tuy nhiên, chúng rất tốn kém về mặt tính toán và khó thực hiện. Một số nghiên cứu [22, 23] chỉ ra rằng các bộ đặc trưng đơn giản vẫn có thể đạt được độ chính xác cao.

Đối với bài toán thực nghiệm này, chúng tôi sử dụng năm đặc trưng cơ bản là Trung bình (*mean*), Độ lệch chuẩn (*standard deviation*), Giá trị hiệu dụng (*RMS*), Trung vị (*median*) và Khoảng biến thiên (*range*). Các đặc trưng này tính toán như trong Bảng II.

5. Phân loại hành động

Chúng tôi sử dụng các bộ phân loại có sẵn trong thư viện Sklearn để thực hiện phân loại hành vi. Bên cạnh độ chính

xác cao và thời gian huấn luyện ngắn, các bộ phân loại còn phải đáp ứng các yêu cầu về thời gian thực. Tập hợp các đặc trưng sẽ được đưa vào quá trình huấn luyện và phân loại. Các phương pháp phân loại được sử dụng bao gồm: Cây quyết định tăng cường Gradient, Máy hỗ trợ vector, Rừng ngẫu nhiên, KNN (k-Nearest Neighbor). Đây cũng là các thuật toán phân lớp được sử dụng rộng rãi trong học máy và phân tích dữ liệu.

Bảng II
TRÍCH CHỌN ĐẶC TRƯNG

Đặc trưng	Định nghĩa
Trung bình (*mean*)	$m = \frac{1}{N}\sum_{i=1}^{N} x_i$ (1)
Độ lệch chuẩn (*standard deviation*)	$\sigma(X_j) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$ (2)
Giá trị hiệu dụng (*RMS*)	$RMS_X = \frac{1}{N}\sum_{i=1}^{N} x_i^2$ (3)
Trung vị (*median*)	$\frac{x_{[\#N/2]} + x_{[\#N/2+1]}}{2}$ (4)
Khoảng biến thiên (*range*)	$[min_{i=1}^{N}\{x_i\}, max_{i=1}^{N}\{x_i\}]$ (5)

Trong đó:

X_j là bản ghi dữ liệu thứ j của trục X

N là số lần lấy mẫu trong mỗi bản ghi

x_i là mẫu thứ i của bản ghi X_j

6. Phép đánh giá

Để xác định tính hiệu quả của thuật toán học máy, chúng tôi biểu diễn kết quả phân loại dưới dạng ma trận nhầm lẫn, với các phép thử được tính như sau:

$$accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \quad (6)$$

$$sensitivity = \frac{TP}{(TP + FN)} \quad (7)$$

$$precision = \frac{TP}{(TP + FP)} \quad (8)$$

Trong đó:

• TP (*True Positive*): Số lượng hoạt động được gán nhãn đúng so với quan sát.

• FN (*False Negative*): Số lượng hoạt động bị gán nhãn sai thành các hoạt động khác.

• FP (*False Positive*): Số lượng hoạt động khác bị gán nhãn sai thành hoạt động đang xem xét.

• TN (*True Negative*): Số lượng hoạt động khác được gán nhãn đúng so với quan sát.

III. KẾT QUẢ VÀ THẢO LUẬN

Kết quả chi tiết của việc phân loại hành vi được trình bày trong Bảng III. Thuật toán phân loại XGBoost cho kết quả tốt vượt trội so với các thuật toán còn lại. Trong bảng ma trận nhầm lẫn ở Bảng IV, hầu hết các mẫu thử nghiệm về hành vi được phân biệt rất tốt (4710/4710 mẫu thử hành động nằm, 3004/3069 mẫu thử hành động đứng, 2777/2843 mẫu thử hành động ngồi, 3561/3596 mẫu thử hành động đi). Nhưng cũng có thể thấy rằng có 95 mẫu được phân loại sai do nhầm lẫn giữa đứng và ngồi, trong khi giữa đứng và đi chỉ có 46 mẫu được phân loại sai, và giữa các hành động khác thì còn thấp hơn.

Bảng III
KẾT QUẢ THỰC HIỆN PHÂN LOẠI HÀNH VI

Thuật toán	Kết quả		
	Accuracy	Sensitivity	Precision
Decision Tree (DT)	91.5%	91%	92%
Gradient Boosted Decision Tree (GBDT)	93%	93%	94%
Support Vector Machine (SVM)	94%	94%	94%
Random Forest (RF)	94%	93%	94%
k-Nearest Neighbor (KNN)	95%	95%	96%
XGBoost	99%	99%	99%

Bảng IV
MA TRẬN NHẦM LẪN

Hành vi quan sát được	Hành vi được dự đoán				Tổng
	Nằm	Đứng	Ngồi	Đi	
Nằm	**4710**	0	0	0	4710
Đứng	0	**3004**	47	18	3069
Ngồi	0	48	**2777**	18	2843
Đi	0	28	7	**3561**	3596
Tổng	4710	3080	2831	3597	14218

Trong cuộc sống hiện đại ngày nay, nhu cầu chăm sóc sức khỏe ngày càng được chú trọng, đặc biệt là người bệnh và người già. Các ứng dụng nhận dạng hoạt động nâng cao được tích hợp để hỗ trợ các hoạt động của con người. Trong đó, việc lựa chọn các đặc trưng đóng một vai trò quan trọng ảnh hưởng tới độ chính xác và tốc độ nhận dạng. Đã có nhiều nghiên cứu lựa chọn khá nhiều đặc điểm để nâng cao tỷ lệ nhận diện hoạt động như [24, 25] sử dụng 43 đặc trưng; [26] sử dụng 64 đặc trưng. Nghiên cứu hiện tại chỉ

sử dụng một bộ đơn giản và hiệu quả gồm năm đặc trưng thống kê và kích thước cửa sổ trượt tối ưu cho hiệu quả nhận dạng tương đối tốt.

Hệ thống của chúng tôi có ưu điểm là độ chính xác cao và có thể chạy trong thời gian thực, do đó có thể được áp dụng cho bệnh nhân hoặc người già. Tuy nhiên nhược điểm dễ nhận thấy đó là hệ thống không thể áp dụng chung cho nhiều đối tượng sử dụng, đặc biệt là những người có cường độ hoạt động cao. Bên cạnh đó, một hạn chế rõ ràng của thiết bị là cần phải có smartphone mang theo và đặt cố định ở tại một vị trí để đảm bảo thuật toán nhận dạng chính xác.

IV. KẾT LUẬN

Kết quả thực nghiệm cho thấy mô hình nhận dạng sử dụng các thuật toán phân loại phổ biến và tập đặc trưng của chúng tôi cho kết quả rất tốt với các hoạt động thường ngày của con người. Việc đơn giản hoá bộ đặc trưng và lựa chọn kích thước cửa sổ tối ưu từ việc phân tích đặc điểm dữ liệu có thể làm cải thiện tốc độ nhận dạng mà vẫn đảm bảo độ chính xác. Ngoài ra, việc lựa chọn một thuật toán phân loại thích hợp cũng ảnh hưởng rõ rệt tới kết quả nhận dạng. Đối với mô hình thử nghiệm của chúng tôi, thuật toán XGBoost giúp cải thiện độ chính xác hơn so với Decision Tree, Gradient Boosted Tree, SVM, RF, KNN. Tuy nhiên, hành vi của con người không chỉ là tự nhiên và tự phát, mà con người có thể thực hiện một số hoạt động cùng một lúc, hoặc thậm chí thực hiện một số hoạt động không liên quan. Đây cũng là thách thức lớn trong bài toán nhận dạng hành động với thời gian thực. Trong tương lai, chúng tôi tiếp tục phát triển hệ thống của mình để có thể dự đoán và xác định các hoạt động đồng thời, hỗ trợ ứng dụng cho các hoàn cảnh sử dụng phức tạp hơn, hướng tới các ứng dụng không chỉ gắn trên người [27], sử dụng những cảm biến nhỏ gọn hơn [28, 29].

LỜI CẢM ƠN

Trần Đức Nghĩa được tài trợ bởi Chương trình học bổng sau tiến sĩ trong nước của Quỹ Đổi mới sáng tạo Vingroup (VINIF), mã số VINIF.2022.STS.38.

Bài báo được hỗ trợ bởi Viện Công nghệ thông tin (IOIT), Viện Hàn lâm Khoa học và Công nghệ Việt Nam (VAST), mã đề tài CSCL02.01/22-23.

TÀI LIỆU THAM KHẢO

[1] N. F. Ghazali, N. Shahar, N. A. Rahmad, N. A. J. Sufri, M. A. As'ari, and H. F. M. Latif, "Common sport activity recognition using inertial sensor," in *2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), Batu Feringghi*, Mar. 2018, pp. 67–71. doi: 10.1109/CSPA.2018.8368687.

[2] F. Hussain, F. Hussain, M. Ehatisham-ul-Haq, and M. A. Azam, "Activity-Aware Fall Detection and Recognition Based on Wearable Sensors," *IEEE Sensors Journal*, vol. 19, no. 12, pp. 4528–4536, Jun. 2019, doi: 10.1109/JSEN.2019.2898891.

[3] A. Vysocky and P. Novak, "HUMAN – ROBOT COLLABORATION IN INDUSTRY," *MM Sci. J.*, vol. 2016, no. 02, pp. 903–906, Jun. 2016, doi: 10.17973/MMSJ.2016_06_201611.

[4] Haibo Ye et al.,"FTrack: Infrastructure-free floor localization via mobile phone sensing," in *2012 IEEE International Conference on Pervasive Computing and Communications*, Lugano, Switzerland, Mar. 2012, pp. 2–10. doi: 10.1109/PerCom.2012.6199843.

[5] W. T. D. souza and K. R, "Human Activity Recognition Using Accelerometer and Gyroscope Sensors," *Int. J. Eng. Technol.*, vol. 9, no. 2, pp. 1171–1179, Apr. 2017, doi: 10.21817/ijet/2017/v9i2/170902134.

[6] B. H. Dobkin, X. Xu, M. Batalin, S. Thomas, and W. Kaiser, "Reliability and Validity of Bilateral Ankle Accelerometer Algorithms for Activity Recognition and Walking Speed After Stroke," *Stroke*, vol. 42, no. 8, pp. 2246–2250, Aug. 2011, doi: 10.1161/STROKEAHA.110.611095.

[7] W.-Y. Lin, V. Verma, M.-Y. Lee, and C.-S. Lai, "Activity Monitoring with a Wrist-Worn, Accelerometer-Based Device," *Micromachines*, vol. 9, no. 9, p. 450, Sep. 2018, doi: 10.3390/mi9090450.

[8] L. Billiet, T. Swinnen, K. de Vlam, R. Westhovens, and S. Van Huffel, "Recognition of Physical Activities from a Single Arm-Worn Accelerometer: A Multiway Approach," *Informatics*, vol. 5, no. 2, p. 20, Apr. 2018, doi: 10.3390/informatics5020020.

[9] Y. Zhu, J. Yu, F. Hu, Z. Li, and Z. Ling, "Human activity recognition via smart-belt in wireless body area networks," *Int. J. Distrib. Sens. Netw.*, vol. 15, no. 5, p. 155014771984935, May 2019, doi: 10.1177/1550147719849357.

[10] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga, "Fusion of Smartphone Motion Sensors for Physical Activity Recognition," *Sensors*, vol. 14, no. 6, pp. 10146–10176, Jun. 2014, doi: 10.3390/s140610146.

[11] L. Gao, A. K. Bourke, and J. Nelson, "Evaluation of accelerometer based multi-sensor versus single-sensor activity recognition systems," *Med. Eng. Phys.*, vol. 36, no. 6, pp. 779–785, Jun. 2014, doi: 10.1016/j.medengphy.2014.02.012.

[12] L. Fan, Z. Wang, and H. Wang, "Human Activity Recognition Model Based on Decision Tree," in *2013 International Conference on Advanced Cloud and Big Data*, Nanjing, China, Dec. 2013, pp. 64–68. doi: 10.1109/CBD.2013.19.

[13] A. Khan, N. Hammerla, S. Mellor, and T. Plötz, "Optimising sampling rates for accelerometer-based human activity recognition," *Pattern Recognit. Lett.*, vol. 73, pp. 33–40, Apr. 2016, doi: 10.1016/j.patrec.2016.01.001.

[14] X. Fafoutis, L. Marchegiani, A. Elsts, J. Pope, R. Piechocki, and I. Craddock, "Extending the battery lifetime of wearable sensors with embedded machine learning," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*, Singapore, Feb. 2018, pp. 269–274. doi: 10.1109/WF-IoT.2018.8355116.

[15] S. Bashir, D. Doolan, and A. Petrovski, "The Effect of Window Length on Accuracy of Smartphone-Based Activity Recognition," *IAENG Int. J. Comput. Sci.*, vol. 43, pp. 126–136, Feb. 2016.

[16] B. Quigley, M. Donnelly, G. Moore, and L. Galway, "A Comparative Analysis of Windowing Approaches in Dense Sensing Environments," *Proceedings*, vol. 2, no. 19, p. 1245, Oct. 2018, doi: 10.3390/proceedings2191245.

[17] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, and I.

Rojas, "Window Size Impact in Human Activity Recognition," *Sensors*, vol. 14, no. 4, pp. 6474–6499, Apr. 2014, doi: 10.3390/s140406474.

[18] G. Alor-Hernández and R. Valencia-García, Current Trends on Knowledge-Based Systems. *Springer International Publishing*, 2017. [Online]. Available: https://books.google.com.vn/books?id=JzFXDgAAQBAJ

[19] M. B. Abidine and B. Fergani, "Evaluating a new classification method using PCA to human activity recognition," in *2013 International Conference on Computer Medical Applications (ICCMA)*, Sousse, Jan. 2013, pp. 1–4. doi: 10.1109/ICCMA.2013.6506158.

[20] T.-P. Kao, C.-W. Lin, and J.-S. Wang, "Development of a portable activity detector for daily activity recognition," in *2009 IEEE International Symposium on Industrial Electronics*, Seoul, South Korea, Jul. 2009, pp. 115–120. doi: 10.1109/ISIE.2009.5222001.

[21] S. J. Preece, J. Y. Goulermas, L. P. J. Kenney, and D. Howard, "A Comparison of Feature Extraction Methods for the Classification of Dynamic Activities From Accelerometer Data," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 3, pp. 871–879, Mar. 2009, doi: 10.1109/TBME.2008.2006190.

[22] M. B. Dehkordi, A. Zaraki, and R. Setchi, "Feature extraction and feature selection in smartphone-based activity recognition," *Procedia Comput. Sci.*, vol. 176, pp. 2655–2664, 2020, doi: 10.1016/j.procs.2020.09.301.

[23] M. Zhang and A. A. Sawchuk, "A bag-of-features-based framework for human activity representation and recognition," in *Proceedings of the 2011 international workshop on Situation activity & goal awareness - SAGAware '11*, Beijing, China, 2011, p. 51. doi: 10.1145/2030045.2030058.

[24] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 74–82, Mar. 2011, doi: 10.1145/1964897.1964918.

[25] C. Catal, S. Tufekci, E. Pirmit, and G. Kocabag, "On the use of ensemble of classifiers for accelerometer-based activity recognition," *Appl. Soft Comput.*, vol. 37, pp. 1018–1022, Dec. 2015, doi: 10.1016/j.asoc.2015.01.025.

[26] G. Vavoulas, C. Chatzaki, T. Malliotakis, M. Pediaditis, and M. Tsiknakis, "The MobiAct Dataset: Recognition of Activities of Daily Living using Smartphones:," in *Proceedings of the International Conference on Information and Communication Technologies for Ageing Well and e-Health*, Rome, Italy, 2016, pp. 143–151. doi: 10.5220/0005792401430151.

[27] Q.-T. Hoang, P. C. Phi Khanh, B. T. Ninh, C. T. Phuong Dung, and T. D. Tran, "Cow Behavior Monitoring Using a Multidimensional Acceleration Sensor and Multiclass SVM," *Int. J. Mach. Learn. Networked Collab. Eng.*, vol. 2, no. 3, pp. 110–118, Sep. 2018, doi: 10.30991/IJMLNCE.2018v02i03.003.

[28] T. D. Tran, D. V. Dao, T. T. Bui, L. T. Nguyen, T. P. Nguyen, and Sugiyama Susumu, "Optimum design considerations for a 3-DOF micro accelerometer using nanoscale piezoresistors," in *2008 3rd IEEE International Conference on Nano/Micro Engineered and Molecular Systems*, Sanya, China, 2008, pp. 770–773. doi: 10.1109/NEMS.2008.4484440.

[29] T.-H. Dao, H.-Y. Hoang, V.-N. Hoang, D.-T. Tran, and D.-N. Tran, "Human Activity Recognition System For Moderate Performance Microcontroller Using Accelerometer Data And Random Forest Algorithm," *EAI Endorsed Trans. Ind. Netw. Intell. Syst.*, vol. 9, no. 4, p. e4, Nov. 2022, doi: 10.4108/eetinis.v9i4.2571.

SƠ LƯỢC VỀ TÁC GIẢ



Nguyễn Quang Huy tốt nghiệp Thạc sĩ ngành Công nghệ Thông tin năm 2017 tại Học viện Công nghệ Bưu chính Viễn thông. Hiện là nghiên cứu viên của Viện Công nghệ Thông tin - Viện Hàn lâm Khoa học và Công nghệ Việt Nam (VAST).
Lĩnh vực nghiên cứu: phân tích dữ liệu, mạng cảm biến không dây và các ứng dụng.
Email: nqhuy@ioit.ac.vn



Trần Đức Nghĩa tốt nghiệp Tiến sĩ Công nghệ thông tin tại trường Université Paris Descartes, Cộng hòa Pháp năm 2018. Hiện công tác tại phòng Tin học quản lý, viện Công nghệ thông tin, VAST.
Lĩnh vực nghiên cứu: phổ EPR, Internet vạn vật, ứng dụng học máy, cảm biến gia tốc.
Email: nghiatd@ioit.ac.vn



Đào Tô Hiệu tốt nghiệp Thạc sĩ ngành Kỹ thuật Điện tử năm 2017 tại Trường Đại học Bách Khoa Hà Nội.
Hiện là giảng viên Khoa Điện-Điện tử, Trường Đại học Phenikaa.
Lĩnh vực nghiên cứu: các thuật toán học máy nhẹ ứng dụng trên vi điều khiển hiệu năng thấp, thiết bị hỗ trợ lính cứu hoả mang trên người, mạng cảm biến, ứng dụng đo lường giám sát IOT.
Email: hieu.daoto@phenikaa-uni.edu.vn



Trần Đức Tân tốt nghiệp Tiến sĩ ngành Kỹ thuật điện tử năm 2010 tại Trường Đại học Công nghệ, Đại học Quốc Gia Hà Nội.
Hiện là giảng viên, Phó trưởng Khoa Điện – Điện tử, Trường Đại học PHENIKAA.
Lĩnh vực nghiên cứu: xử lý tín hiệu, mạng cảm biến không dây và các ứng dụng.
Email: tan.tranduc@phenikaa-uni.edu.vn



Vũ Thị Thương tốt nghiệp Thạc sĩ ngành Khoa học máy tính năm 2009 tại Học viện Kỹ thuật quân sự.
Hiện là giảng viên Khoa CNTT và Truyền thông, Trường Đại học Phương Đông.
Lĩnh vực nghiên cứu: Các phương pháp học máy nhằm cải thiện việc đo tín hiệu, sử dụng cảm biến gia tốc và thuật toán học máy phục vụ bài toán nhận dạng hành vi.
Email: thuongvt@dhpd.edu.vn



Đỗ Xuân Trung tốt nghiệp Thạc sĩ ngành Kĩ thuật Điện tử năm 2014 tại viện Đại học Mở Hà Nội.
Hiện đang công tác tại Khoa CNTT - Điện tử Viễn thông, trường Đại học Kinh Bắc.
Lĩnh vực nghiên cứu: Xử lý tín hiệu và thông tin, thiết kế vi mạch và tích hợp hệ thống.
Email: dotrung@ukb.edu.vn

An Ontology-based Sentiment Analysis Approach to Discovering Hidden Affected Objects

Le Anh Phuong[1], Nguyen Minh Duc[2], Nguyen Dinh Hoa Cuong[3], Nguyen Thi Huong Giang[1]
[1] Department of Computer Science, University of Education, Hue University, Hue
[2] Department of Business Information System, University of Economics, Hue University, Hue
[3] School of Engineering and Technology, Hue University, Hue

Tác giả liên hệ: Nguyen Thi Huong Giang, Email: nthgiang@hueuni.edu.vn
Ngày nhận bài: 29/08/2022, ngày sửa chữa: 17/11/2022, ngày duyệt đăng: 25/11/2022
Định danh DOI: 10.32913/mic-ict-research-vn.v2022.n2.1146

Tóm tắt: Nowadays, the topic of sentiment analysis has attracted a wide range of artificial intelligence technologies such as Natural Language Processing (NLP), neural network. Long Short-Term Memory (LSTM), a deep neural network, has proven the effiency in the task of sentiment classification. However, there is a shortage of semantic knowledge from such classification results. This paper introduces a sentiment analysis approach based on a knowledge model to discover hidden affected objects, which combines ONtology and SentiMent technology, named ONSEM. In order to confirm the efficiency of the ONSEM framework, an experiment on a sentiment analysis dataset was developed and then yielded promising experimental results.

Từ khóa: *Sentiment analysis, ontology, long short-term memory, semantic reasoning, hidden affected object.*

Title: **Phương pháp phân tích quan điểm dựa trên Ontology để khám phá các đối tượng ẩn**

Abstract: Ngày nay, nhiều công nghệ được sử dụng để nghiên cứu chủ đề phân tích quan điểm như trí tuệ nhân tạo, xử lý ngôn ngữ tự nhiên (NLP), mạng nơ ron thần kinh. Bộ nhớ ngắn hạn dài (LSTM), một mạng lưới thần kinh sâu, đã chứng minh sự hiệu quả trong việc phân loại quan điểm. Tuy nhiên, việc dựa vào kết quả phân loại này bị thiếu hụt về kiến thức ngữ nghĩa. Bài báo này giới thiệu một phương pháp phân tích quan điểm dựa trên mô hình tri thức để khám phá những đối tượng ẩn có liên quan, được đặt tên là ONSEM. Để xác nhận sự hiệu quả của ONSEM, một thí nghiệm trên bộ dữ liệu phân tích quan điểm đã được phát triển và đã gặt hái những kết quả đầy triển vọng.

Keywords: *Phân tích quan điểm, Ontology, trí nhớ ngắn hạn dài, suy luận ngữ nghĩa, đối tượng ảnh hưởng bị ẩn.*

I. INTRODUCTION

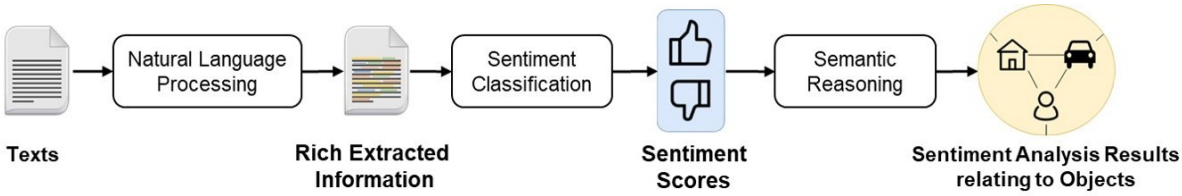
Sentiment analysis employing NLP techniques has been widely applied in many fields, such as market research, finance, politics [1]. In the recent decade, deep learning has reached a big achivement on sentiment classification [2]. Especially, LSTM, an architecture of recurrent neural network (RNN), has shown an outstanding performance because of the ability to handle length texts.

Although LSTM has been adopted in various studies of sentiment classification, the methods of handling knowledge in accordance with sentiment scores were not studied thoroughly. Specifically, related hidden objects affected by sentiment evaluations were not discovered. Ontology, the core of the Semantic Web, has been well-known for the ability of representing knowledge models and semantic reasoning [3]. This study introduces an ontology-based sentiment analysis approach in order to discover objects hiddenly affected by users' reviews, named ONSEM. An experiment on testing the performance of the proposed framework was deployed. The experimental results showed the adequacy of the ONSEM framework.

The rest of this paper is organized as follows: Section II presents relevant studies. Section III introduces the ONSEM framework in detail while the experiment is introduced in Section IV. Lastly, Section V gives the conclusions and suggests future research directions.

II. RELATED WORK

The full elaboration of sentiment analysis studies is out of the scope of this paper. Readers are suggested to find the comprehensively related information in the following survey [4].



Hình 1. Conceptual block diagram of the ONSEM framework.

Sentiment analysis has strong connection to NLP due to the task of preprocessing text data. NLP transforms text to input vectors for classification models. The typical NLP tasks include: to kenization, POS tagging, lemmatization, stemming, stop word removal, and named entity recognition. Typical approaches of sentiment analysis employing NLP can be found in recent studies [5–9].

Within the domain of classification, in recent years, the deep learning approaches have been widely applied to the problem of sentiment analysis because of their high performances of prediction [10]. The literature has been introduced many different deep learning approaches including: (i) weighted Convolutional Neural Networks (CNN) for sentiment analysis of texts [5]; (ii) multiple CNNs composing of CNN, DCNN, MVCNN, and VDCNN for classifying sentiments of Bangla text [6]; (iii) hierarchy CNN for analyzing multi-entity sentiment analysis [7]; (iv) Bi-directional RNN for overcoming the limitations of semantic-information loss and irrelevant shared features [8]; (v) LSTM model for sentiment analysis of hotel reviews [9].

Beside the aformentioned classification models, rule-based classifiers and ontology have also been applied to sentiment analysis. For example, a Semantic Web Rule Language (SWRL) rule-based classifier and ontology was applied to analyzing sentiment of fake news [11]. Sentiment analysis was considered under the view of concept levels with the support of ontology [12]. Sentiment analysis could be used to predict customer interest by combining with e-commerce ontology and fuzzy method [13].

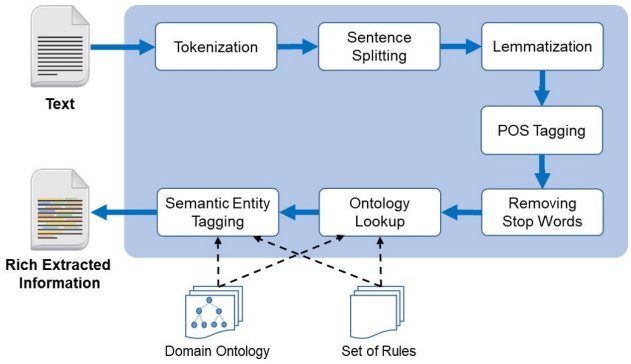
III. METHODOLOGY

1. The ONSEM framework

Figure 1 shows an overview of the ONSEM framework, which consists of three modules including *natural language processing*, *sentiment classification* and *semantic reasoning*. Each of these modules is described in detail as follows.

2. Natural language processing

The first module aims to capture the syntactic and semantic information from the input texts. As shown in Figure 2,



Hình 2. NLP pipeline.

the NLP module includes 7 steps where some steps are supported by NLP tool such as Stanford CoreNLP [14].

Firstly, *tokenization* splits texts into a sequence of tokens, each storing types of text such as word, number, punctuation mark or symbol, etc. Next, *sentence splitting* determines sentences in the texts utilizing the sentence boundary indicators such as periods, question marks, exclamation points. *Lematization* is the task of deriving the root form of each word in inflectional form or in derivational form. The major step is *POS tagging* where each token is labelled with its part-of-speech such as noun, verb, adverb. Insignificant words (e.g. 'a', 'an') are then filtered out by the step of *Removing stop words*. The two last steps utilizes a domain ontology and a set of rules to extract semantic entities from the input texts. While the major information extracted are processed at the next module of *sentiment classification*, the annotation results of semantic entities are further employed at the module of *semantic reasoning*.

3. Sentiment classification

Before being applied sentiment classification, tokens are vectorized by the TF-IDF method [15] (Term Frequency-Inverse Document Frequency) to measure how relevant they are to the input text. The TF-IDF score for the word t in the document d from the set of document D is presented in (1), (2) and (3). Specifically, $freq(t, d)$ is the frequency of the word t in the document d and $count(d \in D : t \in d)$

shows the number of document d in the set of document D containing the word t.

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D) \quad (1)$$

$$tf(t, d) = \log(1 + freq(t, d)) \quad (2)$$

$$idf(t, D) = \log\left(\frac{N}{count(d \in D : t \in d)}\right) \quad (3)$$

In this study, LSTM network [16] was selected for the classification model. Figure 3 shows the LSTM structure at time step t. Specifically, i, o, f denote the input gate, the output gate and the forget gate, respectively. C_t is the memory cell at time t. The inputs of input gate i_t, output gate o_t and forget gate f_t are vector x_t that the network receives at time t and its previous hidden state h_{t-1}.

The LSTM component is formalized as follows:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f), \quad (4)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \quad (5)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \quad (6)$$

$$\hat{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \quad (7)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \hat{c}_t, \quad (8)$$

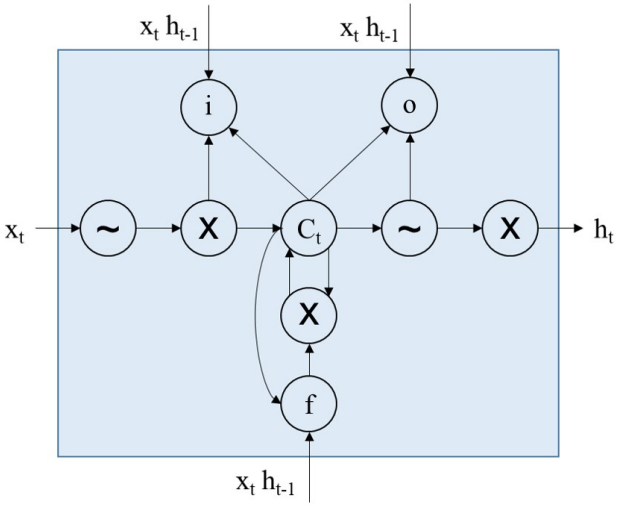
$$h_t = o_t \circ \tanh(c_t), \quad (9)$$

where σ, $\circ$, respectively denote a logistic sigmoid function and element-wise multiplication; f_t, i_t, o_t, c_t, h_t are the forget gate, the input gate, the output gate, the memory cell and hidden state at time-step t. W, U, b are respectively two weights matrices and a bias vector. Within this scope, while the forget gate decides to forget unnecessary information, the input gate directs what new information should be stored in the memory cell. At the final stage, the output gate determines the amount of information in the memory cell that should be exposed. The LSTM model could remember significant information over time steps by this mechanism.

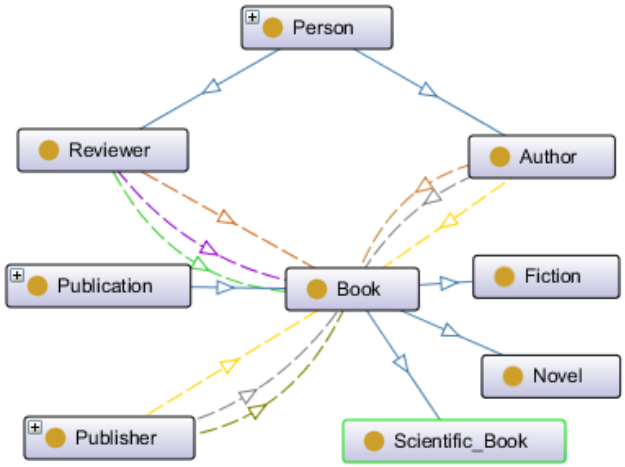
4. Semantic reasoning

The ONSEM semantic reasoning module composes of a domain ontology and its SWRL base. Theoretically, an ontology of domain D is defined as $O_D = \langle C_D, R_D, P_D, I_D \rangle$ where C_D, R_D, P_D, and I_D are the sets of concepts, relations, data properties, and instances, respectively. Within the scope of this paper, a domain ontology of book reviews was developed for the purpose of demonstrating the efficiency of ONSEM framework. An excerpt of this ontology is illustrated in Figure 4.

Based on this ontology, a SWRL rule base was developed in order to discover hidden objects affected by users' reviews. Specifically, results of sentiment analysis found by LSTM model are populated into the ONSEM ontology. Then, the semantic reasoning engine implements its SWRL rules to discover hidden affected objects. Some examples SWRL rules are presented in Table I.



Hình 3. LSTM structure.



Hình 4. An excerpt of the ONSEM ontology.

Bảng I
EXAMPLES OF SWRL RULES

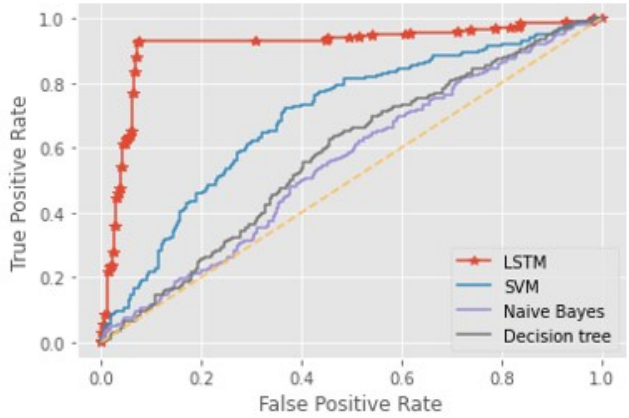
No.	Rule
1	Book(?b) ∧ Author(?a) ∧ Review(?r) ∧ has_Author(?b, ?a) ∧ has_Negative_Review(?b,? r) → negative_Affect_By(?a, ?r)
2	Book(?b) ∧ Author(?a) ∧ Review(?r) ∧ has_Author(?b, ?a) ∧ has_Positive_Review(?b,? r) → positive_Affect_By(?a, ?r)

IV. EXPERIMENT

In order to demonstrate the efficiency of ONSEM framework, a sentiment analysis dataset was retrieved from the Book Genome dataset [17]. Specifically, a tabular dataset containing book title, authors, review text, and rating was

Bảng II
AUC SCORES.

Model	AUC scores
LSTM	**0.914**
SVM	0.670
Naïve Bayes	0.553
Decision tree	0.580

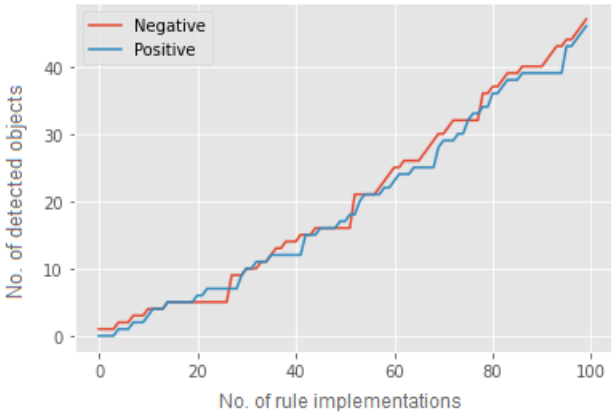


Hình 5. ROC curves of classifiers.

retrieved from JSON-format files of Book Geneom dataset. In which, the original 5-star scale rating was mapped to the binary scale including: (i) negative (1-3 star rating); and (ii) positive (4-5 star rating). Then, any record, which has empty review text, is ignored. Based on this standardized dataset, a 2000-record dataset, which has balance rating labels, was selected and used in our experiments. Additionally, the objects of this dataset (e.g. authors, books) were also populated into the ONSEM ontology as instances. The training set and test set were formed up following the rule of 70%-30%. All of classification models were trained with 10-fold cross-validation method.

The experiments aimed at: (i) confirming the performance of the LSTM model of ONSEM framework; and (ii) demonstrating the reasoning ability of ONSEM's semantic module in discovering objects affected by users' reviews. For the former, the three well-known machine learning models including SVM, Naïve Bayes and decision tree were used as the baseline methods. The performances of these classification models were visualized by ROC curves in Figure 5 and summarized by their AUC scores in Table II.

As showed in Figure 5, the ROC curve of LSTM-based model dominated those of the other models. This domination was also supported by the LSTM AUC score, which is greater than AUC scores of the other classifiers.



Hình 6. Discovering affected objects by reasoning.

The experimental results figured out that LSTM-model outperformed three baseline models.

In order to measure the efficiency of semantic reasoning module, we implemented the process of sentiment analysis and semantic reasoning over the test set and counted the discovered objects affected by users' reviews. The cumulative results of negatively affected objects and positively affected objects are depicted in Figure 6. This promising result confirmed the ability of discovering hidden information by using reasoning process of the ONSEM approach.

V. CONCLUSION

In this study, an ontology-based sentiment analysis approach discovering hidden affected objects was introduced. The ONSEM framework including three modules of natural language processing, sentiment classification and semantic reasoning was described in detail. An experiment on a sentiment analysis dataset was deployed to demonstrate the feasibility of ONSEM framework. For further works, a hybrid approach of LSTM and another deep learning technique will be considered to gain a better result.

TÀI LIỆU THAM KHẢO

[1] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams engineering journal*, vol. 5, no. 4, pp. 1093–1113, 2014.

[2] R. Feldman, "Techniques and applications for sentiment analysis," *Communications of the ACM*, vol. 56, no. 4, pp. 82–89, 2013.

[3] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific american*, vol. 284, no. 5, pp. 34–43, 2001.

[4] M. V. Mäntylä, D. Graziotin, and M. Kuutila, "The evolution of sentiment analysis—a review of research topics, venues, and top cited papers," *Computer Science Review*, vol. 27, pp. 16–32, 2018.

[5] A. Ghorbanali, M. K. Sohrabi, and F. Yaghmaee, "Ensemble transfer learning-based multimodal sentiment analysis using weighted convolutional neural networks," *Information Processing & Management*, vol. 59, no. 3, p. 102929, 2022.

[6] N. R. Bhowmik, M. Arifuzzaman, and M. R. H. Mondal, "Sentiment analysis on bangla text using extended lexicon dictionary and deep learning algorithms," *Array*, vol. 13, p. 100123, 2022.

[7] C. Gan, L. Wang, and Z. Zhang, "Multi-entity sentiment analysis using self-attention based hierarchical dilated convolutional neural network," *Future Generation Computer Systems*, vol. 112, pp. 116–125, 2020.

[8] Y. Cai, Q. Huang, Z. Lin, J. Xu, Z. Chen, and Q. Li, "Recurrent neural network with pooling operation and attention mechanism for sentiment analysis: A multi-task learning approach," *Knowledge-Based Systems*, vol. 203, p. 105856, 2020.

[9] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment analysis using word2vec and long short-term memory (lstm) for indonesian hotel reviews," *Procedia Computer Science*, vol. 179, pp. 728–735, 2021.

[10] J. Joseph, S. Vineetha, and N. Sobhana, "A survey on deep learning based sentiment analysis," *Materials Today: Proceedings*, 2022.

[11] D. Vidanagama, A. Silva, and A. Karunananda, "Ontology based sentiment analysis for fake review detection," *Expert Systems with Applications*, vol. 206, p. 117869, 2022.

[12] A. Khattak, M. Z. Asghar, Z. Ishaq, W. H. Bangyal, and I. A. Hameed, "Enhanced concept-level sentiment analysis system with expanded ontological relations for efficient classification of user reviews," *Egyptian Informatics Journal*, vol. 22, no. 4, pp. 455–471, 2021.

[13] R. Karthik and S. Ganapathy, "A fuzzy recommendation system for predicting the customers interests using sentiment analysis and ontology in e-commerce," *Applied Soft Computing*, vol. 108, p. 107396, 2021.

[14] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.

[15] A. Aizawa, "An information-theoretic perspective of tf–idf measures," *Information Processing & Management*, vol. 39, no. 1, pp. 45–65, 2003.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] D. Kotkov, A. Medlar, A. Maslov, U. R. Satyal, M. Neovius, and D. Glowacka, "The tag genome dataset for books," in *ACM SIGIR Conference on Human Information Interaction and Retrieval*, 2022, pp. 353–357.

AUTHOR'S SUMMARY



Le Anh Phuong is teaching and researching at the Department of Computer Science, Hue University of Education. The author graduated from the Department of Mathematics, Hue University of Education in 1996; received Master and PhD degree in 2001, 2013 at Ha Noi University of Science and Technology. He has been qualified as Associate Professor since 2017. His research interests include Computational Logic, Knowledge Representation and Uncertainty Argument, Hedge Algebra, Descriptive Logic and Semantic Web, Decision Support Systems.

Email: laphuong@hueuni.edu.vn



Nguyen Minh Duc is teaching and researching at the Department of Business Information System, University of Economics, Hue University. The author graduated in 2014 at Department of Economic Information Systems, Hue University; gained Master and PhD degree in 2017, 2021 at Inje University, Korea. His current research focuses are on Data Mining, Text Engineering, Knowledge Management, Ontology, Natural Language Processing.

Email: nguyenminhduc@hueuni.edu.vn



Nguyen Dinh Hoa Cuong is teaching and researching at the School of Engineering and Technology, Hue University. The author graduated in 2002, received Master degree in Informatics in 2006 from University of Science, Hue University and received PhD in Computer Science in 2016 at Khon Kaen University. His research interests are about Data Mining, Text Mining, Knowledge Management, Recomender systems.

Email: ndhcuong@hueuni.edu.vn



Nguyen Thi Huong Giang is teaching and researching at the Department of Computer Science, Hue University of Education. The author graduated in 2002 and received Master Informatics in 2006 from University of Science, Hue University. Her research interests are about Data Mining, Knowledge Management, Ontology.

Email: nthgiang@hueuni.edu.vn

Using Graph Convolution Neural Networks for Solving Mixed integer Linear Programs

Nguyen Thi My Binh[1], Dao Hoang Long[2], Nguyen Van Thien[1], Pham Hong Thai[1]
[1] Hanoi University of Industry
[2] Hanoi University of Science and Technology

Tác giả liên hệ: Nguyen Thi My Binh, Email: binhntm@haui.edu.vn
Ngày nhận bài: 09/08/2022, ngày sửa chữa: 17/11/2022, ngày duyệt đăng: 25/11/2022
Định danh DOI: 10.32913/mic-ict-research vn.v2022.n2.1130

Tóm tắt: This paper focuses on an exact algorithm for solving NP-hard combinatorial optimization problems which frequently requires significant specialized knowledge and trial and error, especially, Mixed Integer Linear Programs (MILP). This challenging, tedious process can be automated by learning the algorithms instead. In many practical applications, the same optimization problem is tackled again and again on regular basis, maintaining the same problem structure but varying in the data. This offers an opportunity for learning algorithms that employ the structure of such recurrent problems. With the motivation, we present a network model for learning branch-and-bound variable selection policies based on reinforcement learning, which leverages the natural variable constraint bipartite graph representation of MILPs. The model is trained going through mimic learning from the strong branching rule. Experimental results claim that using graph convolutional neural networks for solving MILPs can improve for designing branching rules to solve large problems, and outperforms prior models.

Từ khóa: *Graph convolution neural network, mixed integer linear programs, branch and bound.*

Title: **Sử dụng mạng tích chập đồ thị để giải các bài toán quy hoạch nguyên hỗn hợp**

Abstract: Nghiên cứu này tập trung vào giải quyết bài toán quy hoạch nguyên hỗn hợp (MILP), bài toán tối ưu hóa tổ hợp thuộc lớp NP-Hard bằng giải thuật chính xác. Để giải quyết lớp bài toán này, chúng ta cần có những kỹ thuật đặc biệt, thường đòi hỏi chuyên môn cao về kiến thức và thử nghiệm. Quy trình giải các bài toán này có thể được tự động hóa bằng cách sử dụng các thuật toán có khả năng tự học. Trong nhiều ứng dụng thực tế, các bài toán tối ưu hóa được giải quyết giống nhau trên cùng một nền tảng và cấu trúc nhưng thay đổi trong dữ liệu. Điều này cung cấp một cơ hội để học các thuật toán sử dụng cấu trúc của các vấn đề lặp đi lặp lại như vậy. Do đó, chúng tôi trình bày một mô hình mạng cho việc học chính sách phân nhánh cho bài toán nhánh cận dựa trên học tăng cường, tận dụng tích chất có thể biểu diễn dưới dạng đồ thị hai chiều giữa biến và ràng buộc của MILP. Mô hình được huấn luyện thông qua bắt chước học từ quy tắc phân nhánh mạnh. Kết quả thực nghiệm cho thấy sử dụng mạng tích chập đồ thị để giải quyết MILP có thể cải thiện cho vấn đề thiết kế các luật phân nhánh để giải các bài toán với kích thước dữ liệu lớn, và vượt trội so với các mô hình trước đó.

Keywords: *Mạng tích chập đồ thị, quy hoạch nguyên hỗn hợp, nhánh cận.*

I. INTRODUCTION

Combinatorial optimization (CO) problems occurs in various of practical applications, such as scheduling, routing, assignment,cutting and packing, network design, protein alignment, and many other important domains like economic, industrial and scientific. CO problems can be classified into twofolds as discrete and continuous CO problems. Most discrete CO problems (DCOPs) are NP-Hard problems which could not be solved in polynomial time [1]. The available techniques for DCOPs can be categorized into two main: exact and heuristic methods [2].

Mixed-Integer Linear Programs (MILPs) are typical of combinatorial optimization problems. Most CO problems can be formulated as MILPs which receive the attention of many operations researchers. This interest is due in part to the practical importance of problems, but also to its intrinsic difficulty [2]. Several exact algorithms such as branch-and-bound (B&B), dynamic programming, Lagrangian relaxation based methods etc., are guaranteed to find an optimal solution and to prove its optimality for

every instance of MILPs. However, the running execution time often increases significantly with the instance size, and often only small or moderately-sized instances can be tackled to provable optimality. In addition, the ability to execute efficient algorithms is extremely important for solving real-world large-scale combinatorial optimization challenges encountered in many established and emerging heterogeneous areas.

Recently, machine learning (ML) has seen a surge in the development and especially in the deep learning and deep reinforcement learning areas [3]. This has led to significant performance improvements on many tasks within diverse fields. ML accomplishes the imperative require to efficiently solve practical combinatorial optimization problems regarding execution time and quality of the solutions. In many practical application areas we have to access to data of unprecedented scale and accuracy about the system/ process we want to model. Furthermore, ML provides techniques to exploit available data and extract value and useful information, which can be employed for modeling and solving combinatorial problems. This a major driving force for devising innovative solutions to combinatorial challenges. Furthermore, the potential of ML for addressing CO problems is examined by academy community. Recently, novel and advanced techniques for solving CO problems have been strongly developed in ML area [4]. Noticeably, the inherent structure of the CO problems in numerous fields or the data itself is that of a graph [5]. In this paper focuses on applying graph convolution neural network (GCNN) for solving CO problems in generally, mixed-integer linear problems in particularly [6]. This paper focuses on tackling MILPs by exact algorithm known as Branch and Bound (B& B). The idea of B& B algorithm is that it recursively divides the solution space into a search tree, and relaxes bounds along the way to prune subtrees which probably cannot obtain an optimal solution. B& B is a type of tree search algorithms which is widely used tool for solving CO problems. However, in many circumstances it is usually to repeatedly resolver similar combinatorial optimization problems, e.g., scheduling, and day-to-day production planning problems [7], which may strongly differ from the set of instances on which B&B algorithms are typically evaluated. It is necessary to use statistical learning for fine-tune B&B algorithms automatically for a specific class of problems. However, this work exists two challenges. The first one, it is not obvious how to encode the state of a MILP in B&B decision process [8], especially because both search trees and integer linear programs can have a variable structure and size. The second one, it is not show how to formulate a model architecture that derives from rules which can be generalized, at least to similar circumstances. To overcome these shortages, this paper represents how to use graph convolution neural networks for solving MILP problems. More precisely, variable collection in branching problem is considered two aspects as follows:

- How to encode the branching policies into a graph convolution neural network (GCNN), which help with exploit the natural bipartite graph to represent MILP problems.
- Strong branching decisions are approximated by using behavioral cloning with a cross-entropy loss.

The rest of the paper is organized as follows: Related works are presented in Section 2. Background and formulate the branching problem in Section 3. Section 4 presents methodology for solving the branching problem. Experimental results are discussed in Section 5. Finally, Section 6 presents conclusions

II. RELATED WORKS

This section briefly review literature of works that use of machine learning techniques in the context of branching. The authors in [9] focused on variable selection. Their goal is to search a variable selection strategy that intimates the behavior of the classical branching strategy known as strong branching while running faster than strong branching. Alvarez et. al. [8] studied a regression problem and learn directly the strong branching scores of indicates. They then proposed the ExtraTrees which are very robust with respect to the choice of their parameters. Furthermore, in their work, the feature vectors in the training set describe nodes from multiple MILP instances. In [5], the authors was the first proposed a GCNN model for learning greedy heuristics on NP-hard combinatorial optimization problems which are defined on graphs. Selsam et al. in [10], also studied GCNN model, NeuroSAT which can be interpreted a message passing neural network that learns to tackle SAT problem after being trained as classifier to predict satisfiability.

Many works exploited data-driven variable selection from a purely experiment. Karzan et al.[11] divided techniques for picking problem-specific branching rules based upon a partial B&B tree. These branching rules will choose variables that will lead to fast discovering. For CSP tree search, Xia et. al. [12] implemented existing multi-armed bandit algorithms to learning variable selection policies during tree search. In [13], Balafrej et al. applied a bandit approach to choose different levels of propagation during search.

Other works focus on using ML to improve variable selection in B&B, without directly learning a branching policy. The authors in [14] studied how to use ML to determine

an optimal weighting of any set of partitioning procedures for the circumstance distribution at hand using samples. In [15], Liang et al. investigated a variable selection policy for solving SAT problem. They then modeled the variable selection optimization problem as an online multi-armed bandit, a special-case of reinforcement learning, to learn branching variables such that the learning rate of the solver is maximized.

Several works investigate using machine learning techniques in other aspects of B&B beyond variable selection [5, 16, 17]. Authors in [16] focused on learning a node selection heuristic by mimic learning of the expert procedure that widens the node whose feasible set including the optimal solution. In [18], Song et al. studied for node selection and pruning heuristics by mimic learning of shortest paths to obtain good feasible solutions. Khalil et al. [5] presented a theoretical framework for analyzing this decision-making process in a simplified setting, proposed a ML approach for modeling heuristic success likelihood, and designed practical rules that leverage the ML models to dynamically decide whether to run a heuristic at each node of the search tree. Those methods could in principle be combined together to further improve performance for solving combinatorial optimization problems. In addition, many works have divided machine learning approaches to fine-tune exact optimization algorithms.

III. BACKGROUND

1. Mixed integer linear program

We consider a mixed integer linear program which is defined as optimization problem as follows:

$$\arg\min_{x}\{c^T x | Ax \leq b, \mathrm{l} \leq \mathrm{x} \leq \mathrm{u}, \mathrm{x} \in \mathbb{Z}^p \times \mathbb{R}^{n-p}\}, \quad (1)$$

where $c \in \mathbb{Z}^n$ denotes the objective coefficient vector, $A \in \mathbb{R}^{m \times n}$ is the constraint coefficient matrix, $b \in \mathbb{R}^n$ denotes the constraint right-hand-side vector, $l; u \in \mathbb{R}^n$ are the lower and upper variable respectively, $p \leq n$ is the number of integer variables. To solve MILP in (1), the B&B algorithm is applied to integer programming by relaxing the integrality constraint. A continuous linear program (LP) is obtained whose solution by providing a lower bound to MILP in (1). In addition, LP can be tackled efficiently by the simplex algorithm as well. If a solution to the LP relaxation respects the original integrality constraint, then it is also a solution to the problem in (1). If not, then one may decompose the LP relaxation into two sub-problems, by splitting the feasible region according to a variable that does not respect integrality in the current LP solution x^*,

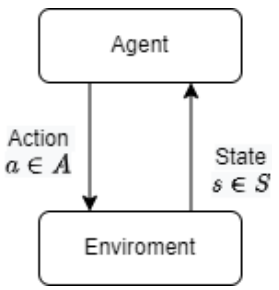
$$x_i \leq \lfloor x_i^* \rfloor \vee x_i \geq \lceil x_i^* \rceil \exists \mathrm{i} \leq \mathrm{p} | \mathrm{x}_i^* \notin \mathbb{Z} \quad (2)$$

where $\lfloor . \rfloor$ and $\lceil . \rceil$ respectively denote the floor and ceil functions. In practice, the two sub-problems will only differ from the parent LP in the variable bounds for x_i, which get updated to $u_i = \lfloor x_i^* \rfloor$ in the left child and $l_i = \lceil x_i^* \rceil$ in the right child.

Applying the B & B algorithm [14], in its simplest formulation, this binary decomposition is performed repeatedly, giving rise to a search tree. The best solution of LP is in the leaf nodes of the tree, which provides a lower bound to the original MILP, whereas the best integral LP solution (if any) provides an upper bound. The stop conditions of B&B algorithm can be obtained whenever both the upper and lower bounds are equal or when the feasible regions do not decompose anymore, as of result providing a certificate of optimality or infeasibility, respectively.

2. The branching problem formulation as a Markov decision process

The B & B algorithm for solving MILP makes the sequential decisions which can be become apart of a Markov decision process [16]. Let the agent be considered the brancher, the solver be the environment. The decision of the solver at the tth and in a state s_t consist of the B & B tree with all past branching decisions. The best integer solution was fathomed so far, the LP solution of each node, the currently focused leaf node, as well as any other solver statistics. The brancher then picks to a variable at among all fractional variables $\mathcal{A}(s_t) \subseteq \{1, 2, ..p\}$ at the currently focused node, according to a policy $\pi(a_t|s_t)$. The solver sequentially extends the B&B tree, tackles the two child LP relaxations, executes any internal heuristic, cuts off the tree if warranted, and finally chooses the next leaf node to split. The algorithm is then in a new state s_{t+1}, and the brancher is called again to take the next branching decision, this progress still executes until the instance is addressed, that is,stop condition of B & B algorithm is no leaf node left for branching. Figure 1 depicts the Markov decision process.



Hình 1. Illustration of the Markov decision process

Episodic is all states that come in from an initial state to terminal state. B & B algorithm is a Markov decision pro-

cess and episode as well. Each episode includes of amount of solving a MILP instanse. Beginning states correspond to an instance initial sampled among a group of interest, while ending states mark the finish of the optimization process. The probability of a trajectory $s = (s_0, s_1, ...s_{\mathcal{T}}) \in \mathcal{T}$ then depends on both the branching policy π and the remaining components of the solver,

$$p_\pi((T)) = p(s_0) \prod_{t=0}^{\mathcal{T}} \sum_{a \in \mathcal{A}(s_t)} \pi(a|s_t) p(s_{t+1}|s_t, a) \quad (3)$$

Reinforcement learning with a carefully designed reward function is a natural approach to search good branching policies.

IV. METHODOLOGY

This section presents methodology for solving MILP by B & B. In details, we use a mimic learning and a GCNN model for B & B algorithm. B & B is type of tree search algorithms. These algorithms recursively partition the search space to find an optimal solution. To keep the tree small, it is so important to carefully decide, when expanding a tree node, which variable to branch on at that node to partition the remaining space. Therefore, variable selection problem (VSP) plays an important role in solving MILP by B & B. VSP can be formulated as a Markov decision process, a natural way of training a policy would be reinforcement learning [19].

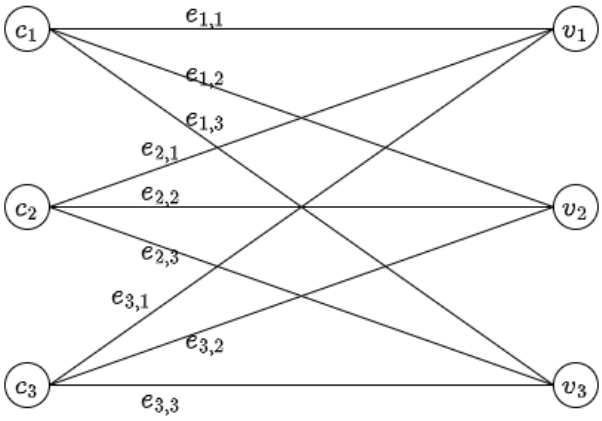
Mimic learning: Model training intimates the strong branching rule [6]. First, the expert on a collection of training instances of interest is run, a dataset of expert state-action pairs is then recorded $Q = \{(s_i, a_i^*)\}$. Second, learn policy should be minimized the cross-entropy loss

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{(s,a_*)} \log_{\pi_0}(a_*|s) \quad (4)$$

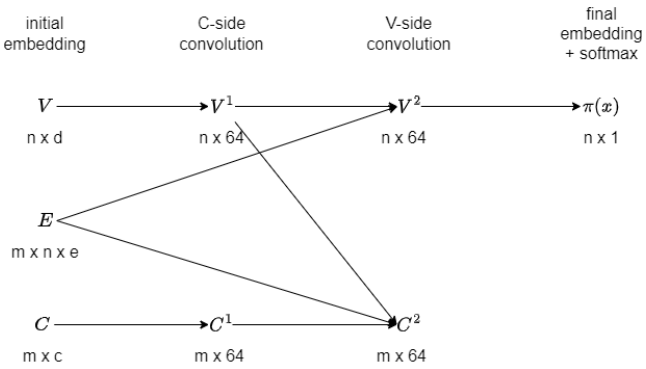
Encoding: the state s_t of the B&B process at time t as a bipartite graph with node and edge features $(G; C; E; V)$ is encoded. Figure 2 demonstrates the graph are nodes corresponding to the constraints in the MILP, in which one per row in the current node's LP relaxation is their feature matrix $C \in \mathbb{R}^{m \times c}$.

Figure 3 illustrates nodes with respect to the variables in the MILP, in which one per LP column, with $V \in \mathbb{R}^{n \times d}$ their feature matrix. An edge (j, j) belongs to E if it links a constraint node i and a variable node j, mean that $A_{ij} \neq 0$ and $E \in \mathbb{R}^{m \times n \times e}$ shows features of edge.

Parametertrization : The variable selection policy $\pi_0(a, s_t)$ is paramerized as a graph convolution neural network (GCNN). To overcome this issue and stabilize the learning procedure, we adopt a simple affine transformation



Hình 2. Illustration of the bipartite state $s_t = (G; C; E; V)$ with $n = 3$ variables and $m = 2$ constraints



Hình 3. Demonstration of the architecture of bipartite GCNN for parametrizing the policy $\pi_\theta(a|s_t)$

$x \leftarrow (x - \beta)/\sigma$, which is *pnorm layer* (The β and σ parameters are initialized with respectively the empirical mean and standard deviation of x on the training dataset)

Model: The base of our GCNN model includes: *pnorm layer*, 2-layer perceptrons with relu activation functions, and bipartite graph convolution

V. EXPERIMENT

1. System and data settings

a) System

We train and evaluate in Ubuntu 20.04.4 (LTS) operating system. This server have configuration:

- Memory: 400GB
- CPU: Intel(R) Xeon(R) Gold 5220 CPU @ 2.20GHz

b) Data settings

The GCNN approach for MILP is evaluated on an NP-hard benchmark as set covering instances which are generated in [20] with 1,000 columns. For more details, each column represents for a variable while each row

represents for a constraint. Training and testing processes are employed on instances with 500 rows. Furthermore, the difficulty of the training set is based on the number of rows, so we evaluate GCNN model on instances with 500 (Easy), 1,000 (Medium) and 2,000 (Hard) rows.

2. Experimental results

GCNN is compared against two prior machine learning branchers as (LMART) [21] and (TREES) [8]. The TREES model uses variable-smart features from bipartite state, obtained by concatenating the variable's node features with edge and constraint node features statistics over its neighborhood. The LMART model uses re-implemented within SCIP. Table I, II and III contain the results about time consumption, a number of wins for each instances, and a

Bảng I
THE EVALUATION OF POLICY ON EASY INSTANCES REGARDING RUNNING TIME (TIME), A NUMBER OF WINS (WINS), AND A NUMBER OF BRANCHING NODES (NODES)

Model	Time	Wins	Nodes
TREES	9.28	0/100	187
LMART	7.19	14/100	167
GCNN	**6.59**	**85**/100	**134**

In Table I, the results show a great challenge which is put on the TREES and LMART models by easy samples. However, GCNN can handle it easily and it outperforms the other algorithms at every index. It takes a high number of wins which is 85 out of 100 instances and also costs the least time and number of branching nodes.

Bảng II
THE EVALUATION OF POLICY ON MEDIUM INSTANCES REGARDING RUNNING TIME (TIME), A NUMBER OF WINS (WINS), AND A NUMBER OF BRANCHING NODES (NODES)

Model	Time	Wins	Nodes
TREES	92.47	0/100	2187
LMART	59.98	0/100	1925
GCNN	**42.48**	**100**/100	**1450**

In Table II, GCNN still performs incredibly. The results show that TREES and LMARTS models totally give up before medium instances. However, GCNN amazingly solves each one of them with lower time and less branching nodes than the rest algorithms. Another notable result is the time, wins and nodes of GCNN against medium samples is higher than the easy one, it makes a hypothesis that the higher branching nodes GCNN creates, the more chances to solve the problem.

Bảng III
THE EVALUATION OF POLICY ON HARD INSTANCES REGARDING RUNNING TIME (TIME), A NUMBER OF WINS (WINS), AND A NUMBER OF BRANCHING NODES (NODES)

Model	Time	Wins	Nodes
TREES	2869.21	0/35	59013
LMART	2165.96	0/54	45319
GCNN	**1489.91**	**66**/70	**29981**

In table III, the results are similar to table I and table II. However, GCNN does not completely beat the hard instances like it has done to the medium ones. It proves that GCNN still has a limit, which means an instances with numerous rows can realize make trouble for GCNN. Therefore, GCNN shows the best result with instances having from 1000 to 2000 rows.

VI. CONCLUSIONS

This paper presents an exact method, brand and bound algorithm for solving mixed integer linear programs as a Markov decision process. By transform the state of branching process into as a bipartite graph, which reduces the required feature engineering by leveraging the variable constraint structure of mixed integer linear programs naturally. We implement experiments on an NP-hard, especially, a set covering problem, by adopting a mimic learning scheme. We then analyze, evaluate a new approach through extensive experiments. The results show that the policies learned of GCNN do better than prior machine learning approaches, even could outperform the standard branching strategy of SCIP. Furthermore, the learned policies of GCNN are able to generalize to instance sizes larger than seen during training. This result is due to collecting strong branching decisions, however, training, can be computationally too expensive on large instances. In short, the GCNN model is better architecture than previous model for the task branching in mixed integer linear program.

ACKNOWLEDGEMENT

This research is funded by Hanoi University of Industry under grant number 25- 2022- RD/HD /- DHCN for Nguyen Thi My Binh.

TÀI LIỆU THAM KHẢO

[1] B. H. Korte, J. Vygen, B. Korte, and J. Vygen, *Combinatorial optimization*, vol. 1. Springer, 2011.
[2] S. Martello, M. Minoux, C. Ribeiro, and G. Laporte, *Surveys in combinatorial optimization*. Elsevier, 2011.
[3] N. Vesselinova, R. Steinert, D. F. Perez-Ramirez, and M. Boman, "Learning combinatorial optimization on graphs: A survey with applications to networking," *IEEE Access*, vol. 8, pp. 120388–120416, 2020.
[4] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: a methodological tour

d'horizon," *European Journal of Operational Research*, vol. 290, no. 2, pp. 405–421, 2021.
[5] E. B. Khalil, B. Dilkina, G. L. Nemhauser, S. Ahmed, and Y. Shao, "Learning to run heuristics in tree search.," in *Ijcai*, pp. 659–666, 2017.
[6] M. Gasse, D. Chételat, N. Ferroni, L. Charlin, and A. Lodi, "Exact combinatorial optimization with graph convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
[7] Y. Pochet and L. A. Wolsey, *Production planning by mixed integer programming*, vol. 149. Springer, 2006.
[8] A. M. Alvarez, Q. Louveaux, and L. Wehenkel, "A machine learning-based approximation of strong branching," *INFORMS Journal on Computing*, vol. 29, no. 1, pp. 185–195, 2017.
[9] E. Khalil, P. Le Bodic, L. Song, G. Nemhauser, and B. Dilkina, "Learning to branch in mixed integer programming," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.
[10] D. Selsam, M. Lamm, B. Bünz, P. Liang, L. de Moura, and D. L. Dill, "Learning a sat solver from single-bit supervision," *arXiv preprint arXiv:1802.03685*, 2018.
[11] F. K. Karzan, G. L. Nemhauser, and M. W. Savelsbergh, "Information-based branching schemes for binary linear mixed integer problems," *Mathematical Programming Computation*, vol. 1, no. 4, pp. 249–293, 2009.
[12] W. Xia and R. Yap, "Learning robust search strategies using a bandit-based approach," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.
[13] A. Balafrej, C. Bessiere, and A. Paparrizou, "Multi-armed bandits for adaptive constraint propagation," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.
[14] M.-F. Balcan, T. Dick, T. Sandholm, and E. Vitercik, "Learning to branch," in *International conference on machine learning*, pp. 344–353, PMLR, 2018.
[15] J. H. Liang, V. Ganesh, P. Poupart, and K. Czarnecki, "Learning rate based branching heuristic for sat solvers," in *International Conference on Theory and Applications of Satisfiability Testing*, pp. 123–140, Springer, 2016.
[16] H. He, H. Daume III, and J. M. Eisner, "Learning to search in branch and bound algorithms," *Advances in neural information processing systems*, vol. 27, 2014.
[17] A. Sabharwal, H. Samulowitz, and C. Reddy, "Guiding combinatorial optimization with uct," in *International conference on integration of artificial intelligence (AI) and operations research (OR) techniques in constraint programming*, pp. 356–361, Springer, 2012.
[18] J. Song, R. Lanka, A. Zhao, A. Bhatnagar, Y. Yue, and M. Ono, "Learning to search via retrospective imitation," *arXiv preprint arXiv:1804.00846*, 2018.
[19] Y. Li, "Deep reinforcement learning: An overview," *arXiv preprint arXiv:1701.07274*, 2017.
[20] E. Balas and A. Ho, *Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study*, pp. 37–60. Berlin, Heidelberg: Springer Berlin Heidelberg, 1980.
[21] C. Hansknecht, I. Joormann, and S. Stiller, "Cuts, primal heuristics, and learning to branch for the time-dependent traveling salesman problem," *arXiv preprint arXiv:1805.01415*, 2018.

AUTHOR'S SUMMARY



Nguyen Thi My Binh is a lecturer of the Faculty of Information Technology, Hanoi University of Industry, Vietnam. She received the M.Sc. degree in Information Technology in 2005, and the PhD. degree in Computer Science at Hanoi University of Science and Technology, Vietnam. Her current research interests include optimization, computational intelligence, Artificial Intelligence.
Email: binhntm@haui.edu.vn or binhntm@fit-haui.edu.vn



Dao Hoang Long graduated Computer Science from Hanoi University of Science and Technology. His research interests include application of heuristic techniques in solving NP-hard problems and applying real-time deep learning in software engineering.
Email: long.dh2000.bachkhoa@gmail.com



Nguyen Van Thien graduated at the Hanoi University of Science and Technology in 1998. He received the M.Sc. degree in Mathematics and Informatics at the Hanoi National University of Education in 2000. He gained the PhD. degree at the Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam. His current research interests include intelligent information system, database, and data mining.
Email: nguyenthien@haui.edu.vn



Pham Hong Thai Pham Hong Thai received Bachelor of Information Technology at the Hanoi University of Industry. He is a Master's student in Information System at Hanoi University of Industry. He is passionate about artificial intelligence.
Email: thaiph99@gmail.com