

KẾT QUẢ ĐỀ XUẤT THUẬT TOÁN PHÁT HIỆN CỘNG ĐỒNG TRÊN MẠNG XÃ HỘI

Nguyễn Hiền Trinh¹, Vũ Vinh Quang^{1*}, Cáp Thanh Tùng²

¹Trường Đại học Công nghệ thông tin và Truyền thông - ĐH Thái Nguyên,

²Trường Đại học Sư phạm - ĐH Thái Nguyên

TÓM TẮT

Hiện nay, phát hiện cộng đồng trên mạng xã hội là một hướng nghiên cứu đang được nhiều nhà khoa học quan tâm. Đã có rất nhiều thuật toán được đề xuất, nhưng một trong những vấn đề cần phải khắc phục đối với các mạng xã hội là trong thực tế số lượng đỉnh và cạnh của đồ thị cực kỳ lớn dẫn tới khối lượng tính toán trong các thuật toán rất lớn, khó đáp ứng được với yêu cầu thực tiễn. Trong bài báo này, chúng tôi đề xuất một giải pháp mới dựa trên tính chất của một số đỉnh đặc biệt có trên đồ thị, từ đó đưa ra thuật toán biến đổi đồ thị ban đầu về dạng rút gọn tương đương nhằm giảm kích thước của đồ thị, đồng thời kết hợp với kỹ thuật lan truyền nhãn, xây dựng hàm heuristic để tăng tốc độ xử lý cho thuật toán phát hiện cộng đồng. Kết quả thực nghiệm trên các bộ dữ liệu chuẩn cho thấy so với phương pháp lan truyền nhãn gốc LPA thời gian xử lý trung bình giảm xuống còn 85,5% trong khi đó chất lượng cộng đồng tăng lên trung bình là 1,145 lần, từ đó đã khẳng định được tính hiệu quả của thuật toán đề xuất.

Từ khóa: Khoa học máy tính; mạng xã hội; cấu trúc cộng đồng; phát hiện cộng đồng; độ đo trung gian đỉnh/cạnh; lan truyền nhãn

Ngày nhận bài: 17/6/2020; Ngày hoàn thiện: 31/8/2020; Ngày đăng: 31/8/2020

THE PROPOSED RESULTS OF ALGORITHM TO DETECT THE COMMUNITY ON SOCIAL NETWORK

Nguyen Hien Trinh¹, Vu Vinh Quang^{1*}, Cap Thanh Tung²

¹TNU - University of Information Technology and Communication,

²TNU - University of Education

ABSTRACT

Nowadays, community detecting on social network has been an orientation which draws attention of many researchers. Numerous algorithms have been proposed, but one of the problems that need to be solved for social networks is the fact that the number of vertices and edges of the graph is extremely large. As a result, the volume of calculations in the algorithms could be significantly large, and thus it is difficult to meet with practical requirements. In this article, we introduce a new approach based on certain attributes of special vertex on network and then propose the algorithm of merging vertices of the same centrality to reduce input vertices/ edges of network, and develop in conjunction with label propagation techniques, build a heuristic function to speed up community detection algorithms. Experimental results on the standard data sets showed that, compared to the original label propagation method LPA, the average processing time was reduced down to only 85.5%, while the quality of the community is increased by an average of 1.145 times, thereby confirming the effectiveness of the proposed algorithm.

Keywords: Computer science; social network; community structure; detecting community structure; the betweenness centrality for vertex/ edges; label propagation

Received: 17/6/2020; Revised: 31/8/2020; Published: 31/8/2020

* Corresponding author. Email: vvquang@ictu.edu.vn

1. Mở đầu

Trong các mạng xã hội, mỗi cộng đồng được coi là một nhóm các đỉnh có thuộc tính chung hoặc có vai trò tương tự. Để nhận biết ra cấu trúc của một cộng đồng, chúng ta dựa vào nhận xét: mối quan hệ giữa các đỉnh trong cùng một cộng đồng là sự gắn kết mạnh, chặt chẽ, dày đặc và ngược lại, với các đỉnh ngoài cộng đồng là sự gắn kết yếu, nhỏ lẻ, rời rạc. Mục tiêu chính của bài toán phát hiện cộng đồng là từ mạng xã hội $G=(V, E)$, chúng ta tìm cách phân lớp các đỉnh trong mạng thành các nhóm tương đương. Trong những năm qua, nhiều phương pháp đã được đề xuất để phát hiện cộng đồng, có thể kể đến các tác giả Flake [1], Radichii [2] đã xây dựng các thuật toán phát hiện cộng đồng theo phương pháp phân đồ thị G thành các đồ thị nhỏ hơn với các đặc trưng riêng; Girvan và Newman [3] đã sử dụng kỹ thuật phân cụm thứ bậc phân chia để phát hiện cộng đồng; Tyler và cộng sự [4]; Pinney và Weshead [5]; Rattigan [6]; Gregory [7] đưa ra các kết quả cải tiến các thuật toán của Girvan và Newman bằng các kỹ thuật tính toán khác nhau. Tuy nhiên, tất cả các thuật toán do các tác giả đề xuất đều có độ phức tạp lớn do phải tính độ đo của tất cả các cạnh trên đồ thị ở mỗi bước xử lý. Hướng cải tiến của chúng tôi là tìm cách giảm chi phí tính toán độ đo cạnh xuống bằng cách nghiên cứu một số đỉnh đặc biệt và tính chất của chúng, từ đó xây dựng phép biến đổi tương đương để giảm số đỉnh và số cạnh của đồ thị, cuối cùng kết hợp với thuật toán lan truyền nhãn trên đồ thị đã được rút gọn để phát hiện cộng đồng nhanh hơn. Hướng cải tiến này sẽ tăng được tốc độ của các thuật toán phát hiện cộng đồng.

Cấu trúc bài báo gồm 5 phần. Phần 1 (mở đầu): giới thiệu về nội dung nghiên cứu. Phần 2: tóm tắt một số kiến thức cơ bản dùng trong mô tả thuật toán. Phần 3: trình bày về lớp đỉnh treo tương đương, thuật toán kết hợp, phép biến đổi tương đương và đồ thị rút gọn. Phần 4: phát triển thuật toán gán nhãn nhằm phát hiện cộng

đồng thực hiện trên đồ thị rút gọn. Phần 5: đưa ra các kết quả thực nghiệm kiểm tra các thuật toán đề xuất trên các bộ dữ liệu.

2. Một số kiến thức cơ bản

2.1. Đồ thị mạng xã hội: Là cấu trúc $G=(V, E)$, trong đó V là tập các đỉnh và E là tập các cạnh, trong đó tập cạnh E thể hiện mối quan hệ xã hội giữa các thành viên với nhau [8].

Đồ thị G có thể là vô hướng hoặc có hướng tùy theo mối quan hệ giữa các thành viên. Cấu trúc cộng đồng C được định nghĩa là tập con các đỉnh của V sao cho với mỗi đỉnh $v \in C$ có nhiều cạnh kết nối v với những đỉnh khác trong C và ít cạnh kết nối v với những đỉnh khác thuộc $V \setminus C$ [8]. Ta ký hiệu δ_{st} là số đường đi ngắn nhất đi từ s tới t ; $\delta_{st}(v)$ là số đường đi ngắn nhất đi từ s tới t và có đi qua v ;

$\tau_{st}(v) = \frac{\delta_{st}(v)}{\delta_{st}}$ là độ phụ thuộc của cặp đỉnh

s, t vào đỉnh v . Không làm mất tính tổng quát, giả thiết G là đồ thị vô hướng và áp dụng một số tính chất về đường đi trong đồ thị G [9], khi đó các cạnh và các đường đi luôn có tính đối xứng nghĩa là:

$$(u, v) \in E \text{ thì } (v, u) \in E,$$

$$\delta_{st} = \delta_{ts}, \delta_{st}(v) = \delta_{ts}(v), \tau_{st}(v) = \tau_{ts}(v).$$

2.2. Độ đo trung gian (The betweenness centrality): là độ đo của một đối tượng v trong mạng G được xác định bằng tổng số các đối tượng khác trong mạng có thể trao đổi và liên kết với nhau thông qua đối tượng đó trên G [9]. Độ đo trung gian được xác định bằng công thức:

$$C_B(v) = \sum_{s \neq t \neq v} \tau_{st}(v) = \sum_{s \neq t \neq v} \frac{\delta_{st}(v)}{\delta_{st}} \quad (1)$$

Trong đó, nếu $s = t$ thì $\delta(s, t) = 1$,

nếu $v = s$ hoặc $v = t$ thì $\delta(s, t) = 0$,

$$\delta(s, t) = \sum_{u \in N(t)} \delta(s, u) \text{ trong đó}$$

$$N(t) = \{u : (u, t) \in E; d(s, t) = d(s, u) + 1\};$$

$d(s, t)$ là khoảng cách giữa hai đỉnh s và t .

Trong công thức (1), do tính chất đối xứng của các đường đi, ta chỉ cần tính một chiều từ s đến t . Giá trị $C_B(v)$ có ý nghĩa là hệ số tiềm năng để điều khiển sự liên kết giữa các đỉnh trên đồ thị. Nói cách khác, độ trung gian của đỉnh v là tổng các xác suất những đường đi ngắn nhất đi qua v trên tất cả các cặp đỉnh có thể của đồ thị G .

2.3. Độ trung gian của cạnh: $C_B(e)$ là độ đo trung gian của cạnh $e = (a, b)$ trên đồ thị mạng G được định nghĩa bằng tỷ số các cặp đỉnh x và y mà cạnh e nằm trên đường đi ngắn nhất giữa x và y , với mọi $x, y \in V$ [9]. Độ đo trung gian được xác định bằng công thức:

$$C_B(e) = \sum_{s \neq t} \frac{\delta_{st}(e)}{\delta_{st}} \quad (2)$$

Do tính đối xứng nên độ đo trung gian của cạnh e theo công thức (2) sẽ bằng hai lần tổng của $\tau_{st}(e)$ tính theo một chiều với $s \neq t \in V$.

Để tính độ đo trung gian của các đỉnh và các cạnh trên G , ta thường sử dụng phương pháp duyệt theo chiều rộng BFS bắt đầu từ đỉnh x bất kỳ, kết quả sẽ đưa ra một đồ thị định hướng phi chu trình DAG_x (Directed Acyclic Graph) trong đó mỗi cạnh của DAG_x sẽ là một phần của một đường đi ngắn nhất xuất phát từ x .

2.4. Tiêu chuẩn đánh giá cộng đồng

Newman và Girvan [10], X. Liu và cộng sự [11] đã đưa ra đại lượng Modularity và tối ưu hóa để đánh giá chất lượng cộng đồng:

$$Q = \frac{1}{2m} \sum_{ij} (a_{ij} - p_{ij}) \delta(C_i, C_j) \quad (3)$$

Trong đó, $A = (a_{ij})_{n \times n}$ là ma trận kề, $2m = \sum_{ij} a_{ij}$, p_{ik} là số cạnh dự kiến có trong C , hàm $\delta(C_i, C_j) = 1$ nếu i, j thuộc cùng một cộng đồng và $\delta(C_i, C_j) = 0$ nếu ngược lại. Dựa trên xác suất kết nối giữa đỉnh i và đỉnh j ta có:

$$Q = \frac{1}{2m} \sum_{ij} (a_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j) \quad (4)$$

với k_i, k_j là bậc của đỉnh i và đỉnh j . Nếu kí hiệu n_c là số cộng đồng, l_c là số cạnh nối các đỉnh, d_c là tổng số bậc của các đỉnh của cộng đồng C . Khi đó ta có:

$$Q = \sum_{c=1}^{n_c} (\frac{l_c}{m} - (\frac{d_c}{2m})^2) \quad (5)$$

Một cộng đồng C có đại lượng Q đạt giá trị dương và càng lớn thì cộng đồng được xác định càng rõ, tức là việc phân cộng đồng là tốt.

3. Lớp đỉnh treo tương đương

3.1. Đỉnh treo và lớp đỉnh treo tương đương

Định nghĩa 1: Cho đồ thị mạng $G = (V, E)$, đỉnh $v \in V$ được gọi là đỉnh treo nếu bậc của nó là 1. Xét W là hàm trọng số của các đỉnh thuộc G . Khi đó G có dạng $G = (V, E, W)$. Với đồ thị mạng $G = (V, E, W)$ vô hướng liên thông, u và $w \in V$ là hai đỉnh treo, u tương đương bậc 1 với w , ký hiệu $u \approx_1 w$ khi và chỉ khi chúng cùng liên kề với v , $(N(u) = N(w) = \{v\})$.

Dễ nhận thấy quan hệ $\approx_1$ là quan hệ tương đương. Ký hiệu V_T là tập tất cả các đỉnh treo của đồ thị G , $V_T = \{u \in V \mid \deg(u) = 1\}$, tập các đỉnh $V_T / \approx_1$ sẽ tạo ra các lớp của những đỉnh treo tương đương với nhau, $V_T / \approx_1 = \{C_1, C_2, \dots, C_k\}$. Những đỉnh treo tương đương với nhau sẽ cùng liên thuộc với một đỉnh và có cùng một độ đo trung gian. Từ đây ta có thể kết hợp tất cả những đỉnh treo tương đương của lớp $C_i, i = 1..k$ thành đỉnh đại diện C'_i (cũng là đỉnh treo), ta nhận được đồ thị $G_1 = (V_1, E_1, W_1)$, trong đó:

$$V_1 = V - V_T \cup V_C,$$

$$V_T = \{u \in V \mid \deg(u) = 1\};$$

$$V_C = \{C'_1, C'_2, \dots, C'_k\},$$

$$E_1 = E - \{(u, v) \mid u \in V_T, v \in N(u)\} \cup \{(v, C'_i) \mid i = 1..k, v \in N(u), u \in C_i\}$$

$$W_1(v) = 1 \text{ với}$$

$$v \in V - V_T; W_1(v) = |C_i|, v \in V_C$$

Xuất phát từ khái niệm đỉnh treo, lớp đỉnh treo tương đương, chúng ta nhận được các tính chất sau:

Tính chất 1. Mọi đỉnh treo của đồ thị G hoặc là gốc hoặc là lá trên các DAG được duyệt theo BFS.

Tính chất 2. Nếu v là đỉnh treo của đồ thị G và $e = (v, w) \in E$ thì $C_B(e) = (|V| - 1)$.

Tính chất 3. Nếu v là đỉnh treo của đồ thị G và w là đỉnh liền kề với v , $(v, w) \in E$ thì DAG_v và DAG_w cùng có chung một đồ thị con $G_1 = DAG_v \cap DAG_w$.

Tính chất 4. Với mọi đỉnh treo $u \in V$, $v \in V$ là đỉnh liền kề với đỉnh u ($u \in P_u(v)$). Khi đó ta có các hệ thức:

$$(i) \quad \delta_{ut} = \delta_{vt} \text{ với mọi } t \in V - \{u, v\}$$

$$(ii) \quad \delta_{ut}(w) = \delta_{vt}(w) \text{ với mọi } w \in V - \{s \in V | \deg(s) = 1\}, \\ t \in V - \{u, v, w\}$$

$$(iii) \quad \tau_{ut}(v) = 1 \text{ với mọi đỉnh } t \in V - \{u, v\}$$

$$(iv) \quad C_B(v) = |N_1(v)| * \left(\frac{|N_1(v)| - 1}{2} + \sum_{s \neq v, t \in V - N_1(v)} \frac{\delta_{st}(v)}{\delta_{st}} \right)$$

Tính chất 5. Giả sử G_1 là đồ thị rút gọn của đồ thị G sau khi kết hợp các đỉnh treo tương đương, V_C là tập các đỉnh treo của G_1 . Khi đó ta có:

$$(i). \tau_{st}(v) = W_1(C'_i) * \tau_{ut}(v) \text{ nếu } s = C'_i, i = 1..k, u = N(C'_i), v \neq u, \\ t \in V_1 - V_C - \{u\},$$

$$(ii). \tau_{st}(v) = W_1(C'_i) * \frac{(W_1(C'_i) - 1)}{2}, \text{ nếu } s = C'_i, t = C''_i, C'_i, C''_i \in C_i, i, j = 1..k, \\ v = N(C'_i),$$

$$(iii). \tau_{st}(v) = W_1(C'_i) * W_1(C'_j) * \tau_{vw}, \text{ nếu}$$

$$s = C'_i, t = C'_j, i, j = 1..k, i \neq j,$$

$$v = N(C'_i), w = N(C'_j),$$

$$(iv). \tau_{st}(v) = W_1(C'_i) * W_1(C'_j) * \tau_{uw}(v) \text{ nếu}$$

$$s = C'_i, t = C'_j, i, j = 1..k, i \neq j,$$

$$u = N(C'_i), w = N(C'_j), v \in V_1 - V_C - \{u\},$$

$$(v). \tau_{st}(v) = W_1(C'_i) * \tau_{sw}(v) \text{ nếu}$$

$$s \notin V_C, t = C'_i, i = 1..k, w = N(C'_i),$$

$$v \in V_1 - V_C.$$

Chứng minh:

Tính chất 1: Đỉnh trên DAG mà không có đường đi nào qua nó được gọi là lá. Dễ nhận thấy, các đỉnh treo nếu không phải là đỉnh gốc của DAG thì phải là lá.

Tính chất 2: Vì G là liên thông nên với mọi $v' \in V - \{v\}$ đều có đường đi tới v . Vì v là đỉnh treo nên mọi đường đi ngắn nhất giữa chúng đều đi qua e . Theo định nghĩa, độ trung gian của cạnh sẽ là $C_B(e) = |V| - 1$.

Tính chất 3: Hiển nhiên đồ thị G_1 chính là đồ thị con của DAG_v (hoặc DAG_w) thu được bằng cách bỏ đi các đỉnh treo liên thuộc với w cùng cạnh nối giữa chúng.

Bằng lập luận tương tự, ta chứng minh được các tính chất còn lại.

3.2. Thuật toán kết hợp các đỉnh treo tương đương

Xuất phát từ khái niệm và các tính chất về lớp các đỉnh treo, chúng tôi đề xuất thuật toán kết hợp các đỉnh treo tương đương nhằm giảm số đỉnh của đồ thị ban đầu G về đồ thị rút gọn G_1 .

Thuật toán 1 (Kết hợp các đỉnh treo tương đương)

Input : Mạng $G = (V, E)$

Output: $G_1 = (V_1, E_1)$ // đồ thị thu được sau khi kết hợp các đỉnh treo tương đương

Begin

```

for  $u \in V$  do  $W(u)=1$  // Xác định  $W$ 
 $V_1 = V$ ;  $E_1 = E$ ;  $P = \phi$ ;
for  $u \in V$  do
if ( $\deg(u) == 1$  &  $\&(u, v) \in E$ ) then
    {  $P.push(u, v)$ ;
       $V_1 = V_1 - \{u\}$ ;  $E_1 = E_1 - \{(u, v)\}$  }
 $k = 1$ ; //Tìm các lớp đỉnh treo tương đương
 $(u, v) = P.pop()$ ;  $C[k] = \{u\}$ ;  $N[k] = v$ ;
while ( $P \neq \phi$ ) do
    {  $(u, v) = P.pop()$ ;  $j = 1$ ; loop = true;
      while ( $j \leq k$  &  $\&$  loop) do
          if ( $N[j] == v$ ) then
              {  $C[j] = C[j] \cup \{u\}$ ; loop=false } else  $j = j + 1$ ;
            if (loop) then {  $k = k + 1$ ;
                           $C[k] = \{u\}$ ;  $N[k] = v$  } }
     $V_c = \phi$ ;
    //Kết hợp các đỉnh treo tương đương
    for  $j = 1$  to  $k$  do {  $V_c = V_c \cup \{C'_j\}$ ;
                         $W_1(C'_j) = |C[j]|$ ;  $E_1 = E_1 \cup \{(C'_j, N[j])\}$  }
    for  $u \in V_1$  do
         $W_1(u) = W(u)$ ;  $V_1 = V_1 \cup V_c$ ;

```

End.

Nhận xét: Thuật toán cần duyệt qua tất cả các đỉnh của đồ thị để xác định tập các đỉnh treo (vòng lặp thứ 2), do đó độ phức tạp của thuật toán là $O(n)$ với n là số đỉnh của mạng.

4. Phát triển thuật toán gán nhãn để phát hiện cấu trúc cộng đồng

4.1. Giới thiệu về thuật toán LPA

Thuật toán gán nhãn (LPA) là một thuật toán được đề xuất bởi Raghavan và các cộng sự [12]. Ý tưởng cốt lõi của LPA là gán từng nút trong mạng vào một cộng đồng, nơi mà hầu hết các nút láng giềng của nó sẽ quyết định nó

có thuộc về cộng đồng mới hay không. Cụ thể, cho một mạng có n nút, trong đó nút x có nhãn ở lần lặp t được ký hiệu là $Cx(t)$, LPA được mô tả như sau:

1. Khởi động nhãn ban đầu $Cx(0) = x$.
2. Đặt $t = 1$.
3. Lấy danh sách các nhãn được cập nhật cho các nút theo thứ tự ngẫu nhiên.
4. Đối với mỗi nút trong danh sách, xác định $Cx(t) = f(Cx_1(t), \dots, Cx_m(t), Cx_{m+1}(t-1), \dots, Cx_n(t-1))$.

(Trong đó $Cx_1(t), \dots, Cx_m(t)$ là nhãn của các nút lân cận đã được cập nhật trong lần lặp hiện tại, còn $Cx_{m+1}(t-1), \dots, Cx_n(t-1)$ là nhãn của các nút lân cận trong lần cập nhật trước. Hàm f trả về nhãn được chấp nhận bởi hầu hết các nút lân cận của nó.)

5. LPA đạt đến sự hội tụ nếu mọi nút có nút láng giềng là nhiều nhất có thể có), thì dừng thuật toán. Nếu không, đặt $t = t + 1$ và quay lại bước (3)

6. LPA dừng nếu đạt được số lần hội tụ hoặc số lần lặp tối đa do người dùng định nghĩa.

Nhận xét: LPA là một thuật toán heuristic, chạy nhanh và cho kết quả khá tốt đối với các tập dữ liệu lớn [13]. LPA sẽ phụ thuộc vào việc chọn hàm f là hàm heuristic, giá trị của hàm phụ thuộc vào trạng thái hiện tại của bài toán tại mỗi bước giải và đưa ra tính toán để xác định nhãn hợp lý trong từng bước của thuật giải [14], [15]. Mặc dù chưa khẳng định về tính ổn định và độ chính xác, song thuật toán LPA được đánh giá là một thuật toán đơn giản và hiệu quả đối với bài toán xác định cấu trúc cộng đồng trên mạng xã hội [16], [17].

Từ kết quả nghiên cứu lớp các đỉnh treo tương đương và LPA, chúng tôi đề xuất thuật toán mới phát hiện cộng đồng mới dựa trên việc rút gọn các đỉnh treo và phát triển thuật toán lan truyền nhãn trên tư tưởng gán nhãn cho các nút chưa được gán nhãn trước đó trên đồ thị đã được rút gọn.

4.2. Đề xuất thuật toán 2: LPA_D (LPA_Degree)

Như chúng ta đã biết, nhãn của nút thứ i , kí hiệu là C_i được xác định bởi công thức tổng quát:

$$C_i = f_i = f(C_1(t), C_2(t), \dots, C_m(t), C_{m+1}(t-1), C_{m+2}(t-1), \dots, C_k(t-1)) \quad (6)$$

Trong đó, hàm f là hàm Heuristic cần xác định. Trên cơ sở của thuật toán LPA, chúng tôi sẽ lựa chọn hàm f theo một trong những quan điểm đã được đưa ra để đánh giá xu thế hình thành cộng đồng cho mạng xã hội. Theo Phillip Bonacich [18] và Linton Freeman [19] thì độ đo trung tâm của một đỉnh v ($\text{degree}(v)$ = tổng số cạnh kết nối với đỉnh v) là trị số nói lên vai trò quan trọng, mức độ ảnh hưởng của đỉnh đó đối với các láng giềng nói riêng và với cộng đồng mà nó thuộc về nói chung. Hiểu theo nghĩa nào đó thì một đỉnh được coi như một tác nhân và tác nhân càng có nhiều mối quan hệ (quan hệ rộng) thì nó càng quan trọng đối với cộng đồng (sức lan tỏa và ảnh hưởng của cá nhân đó với cộng đồng càng lớn, sức thu hút để hình thành nên cộng đồng càng cao v.v...). Như vậy, theo quan điểm đó thì độ đo trung tâm sẽ đóng vai trò quan trọng nhất trong việc xác định cộng đồng. Đề thiết kế thuật toán, chúng tôi cũng sẽ đặt trọng tâm vào tham số này vì tính thực tiễn, đơn giản và hiệu quả của thuật toán khi cần xử lý với dữ liệu lớn. Do vậy, hàm gán nhãn mà chúng tôi lựa chọn khi cài đặt thuật toán được xác định bằng công thức:

$$C_i = f_i = f_{\text{Degree}} = \text{Max}_{j \in N(i)} \{ \text{degree}(j) \} \quad (7)$$

Như vậy, tại mỗi vòng lặp của quá trình lan truyền nhãn, nút i sẽ lựa chọn theo xu thế ngã về người láng giềng có vai trò quan trọng nhất (có trị số degree lớn nhất). Nếu có nhiều hơn một láng giềng có degree lớn nhất bằng nhau, nút i sẽ lấy ngẫu nhiên theo một trong số đó. Nếu không ai tốt hơn nó, nó sẽ không đổi nhãn (tức là nó chính là vai trò trung tâm trong giai đoạn đang xét). Vòng lặp sẽ tiếp tục cho đến khi đạt được trạng thái hội tụ. Trên cơ sở đó, thuật toán đề xuất được mô tả như sau:

Algorithm LPA_D

Input: Mạng $G=(V, E)$, biểu diễn bởi ma trận kề A ;

Output: Tập các cộng đồng của mạng.

Begin

1. Kết hợp các đỉnh treo tương đương (Thuật toán 1) thu được đồ thị rút gọn $G_1 = (V_1, E_1)$
2. For each $x \in V_1$, $C_x(0) = x$; // Khởi tạo mỗi nút là một cộng đồng, thể hiện bằng nhãn của nó.
3. Vòng lặp lan truyền nhãn
 - 3.1. $t = 1$;
 - 3.2. Tạo tập X gồm các nút theo thứ tự ngẫu nhiên
 - 3.3. For each $x \in X$
 - 3.3.1. For $y \in N(x)$ tính $C_x = f_x = f_{\text{Degree}} = \text{Max}_{y \in N(x)} \{ \text{degree}(y) \}$
 - 3.3.2. Cập nhật nhãn của x theo y , xác định cộng đồng mà x, y thuộc về.
 - 3.4. Nếu mọi nút có cùng nhãn với các láng giềng thì dừng thuật toán.
Nếu không, đặt $t = t + 1$ và quay lại bước (3.2).
4. Xác định kết quả:
 - 4.1. Cộng đồng gồm các node có cùng nhãn
 - 4.2. Số lượng nhãn (kiểu của nhãn) cho biết số lượng cộng đồng.

End.

Độ phức tạp thời gian của LPA_D được đánh giá bằng $\text{Max}\{O(k_1n), O(k_2n)\}$ với n là số đỉnh của G_1 , k_1 là số láng giềng lớn nhất của đỉnh trong G_1 và k_2 là số lần lặp lan truyền nhãn ($k_i \geq 1, i=1,2$).

Nhận xét

+ Trong thuật toán LPA_D, chúng ta đã sử dụng thuật toán kết hợp các đỉnh treo tương đương ngay tại bước đầu tiên, do đó hiển nhiên ta thu được đồ thị rút gọn $G_1 = (V_1, E_1)$ với số đỉnh nhỏ hơn nhiều so với đồ thị ban đầu $G=(V, E)$, do đó độ phức tạp của thuật toán lan truyền nhãn áp dụng với đồ thị rút gọn sẽ giảm so với độ phức tạp của thuật toán áp dụng cho đồ thị ban đầu.

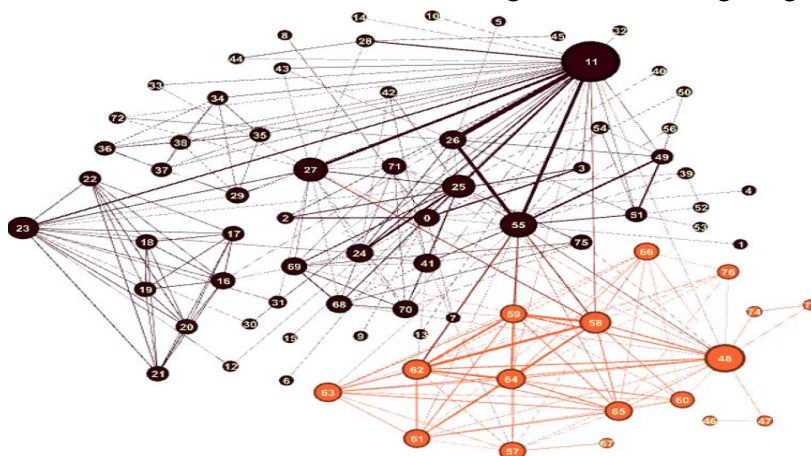
+ Theo phân tích về ý nghĩa của độ đo trung tâm, trong thuật toán đã sử dụng hàm gán nhãn theo công thức (7), khi đó chất lượng cộng đồng thu được từ thuật toán LPA_D chắc chắn sẽ tốt hơn so với việc sử dụng các hàm gán nhãn khác.

5. Một số kết quả thực nghiệm

Để kiểm tra tính hiệu quả của thuật toán đã đề xuất, chúng tôi tiến hành thực hiện thuật toán trên một số bộ dữ liệu thực giả định với kích thước lớn cũng như một số bộ dữ liệu mẫu tham khảo trong các thư viện quốc tế đã được công bố. Trong các thực nghiệm này, chúng tôi tiến hành so sánh về thời gian chạy cũng như chất lượng cộng đồng của thuật toán đề xuất với thuật toán LPA gốc. Các chương trình được thực hiện trên môi trường Matlab 2012.

Ví dụ 5.1. Les Misérables Group: Năm 1993, nhà nghiên cứu D. E. Knuth của đại học

Stanford (Mỹ) đã xây dựng bộ dữ liệu dựa vào tác phẩm *Những người khốn khổ* của Đại văn hào Victor Hugo, mô tả mối quan hệ giữa các nhân vật (có 77 node tương ứng với 77 nhân vật, nếu nhân vật nào cùng xuất hiện trong một chương (chapter) thì hai node đó có cạnh nối). Do vậy đồ thị có 77 node và 254 cạnh. (Dữ liệu lấy từ: <https://github.com/gephi/gephi/wiki/Datasets>). Chúng tôi đã sử dụng bộ dữ liệu đó và thực hiện LPA_D, chỉ với 6 vòng lặp, thu được hai cộng đồng với các đỉnh được mô tả như trong hình 1 (Đồ thị biểu diễn bởi phần mềm Gephi 0.9.2). Qua hình biểu diễn, dễ dàng nhận ra hai nhân vật quan trọng trong mạng là 2 nhân vật có ảnh hưởng lớn đến cộng đồng (có degree cao), đó là nút số 11 (Valjean) và số 48 (Gavroche). Có thể xem đó chính là 2 nút trung tâm tạo ra 2 cộng đồng.



Hình 1. Kết quả các cộng đồng

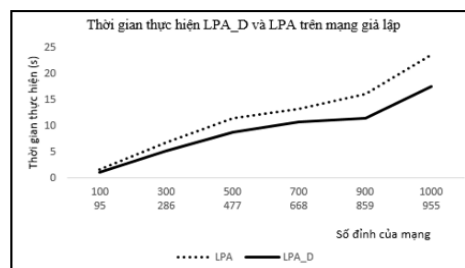
Ví dụ 5.2: Xét 6 bộ dữ liệu giả định với kích thước lớn, tiến hành thực nghiệm thuật toán LPA_D và so sánh với thuật toán LPA [12]. Nhóm tác giả thu được kết quả trong bảng 1 và hình 2.

Bảng 1. Kết quả so sánh hai thuật toán

(*n*-số đỉnh, *m*-số cạnh, *k*-số đỉnh treo, *l*-số lớp đỉnh treo tương đương, *t*-thời gian)

(Đơn vị tính thời gian chạy:s)

Stt	n	m	k	l	t_LPA	t_LPA_D
1	100	500	15	5	1,7	1,1
2	300	1500	45	14	6,8	5,2
3	500	2500	75	23	11,5	8,7
4	700	3500	105	32	13,3	10,7
5	900	4500	135	41	16,1	11,5
6	1000	5000	150	45	23,6	17,6

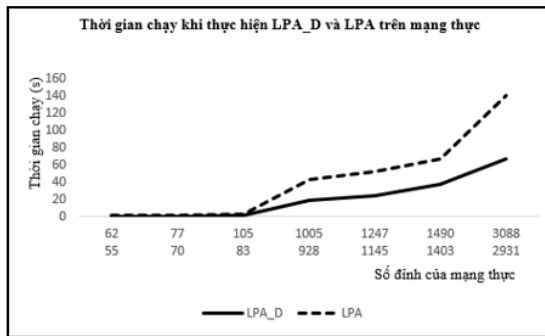


Hình 2. So sánh thời gian thực hiện

Ví dụ 5.3: Tiếp tục xét các bộ dữ liệu chuẩn được công bố trong thư viện Quốc tế [20], chúng tôi tiến hành thử nghiệm thuật toán, đánh giá về kích thước mạng, so sánh thời gian chạy, tính toán chất lượng cộng đồng thu được và so sánh với LPA [12], kết quả được đưa ra trong bảng 2 và các hình 3 và hình 4.

Bảng 2. Kết quả thực nghiệm trong ví dụ 5.3 (*n*-số đỉnh, *m*-số cạnh, *k*-số đỉnh treo, *l*-số lớp đỉnh treo tương đương, *L*- số lần lặp, *c*-số cộng đồng, *Q*-chất lượng, *t*-thời gian)
(Đơn vị tính thời gian chạy:s)

STT	Bộ dữ liệu	n	m	k	l	c	L	Q		t	
								LPA_D	LPA	LPA_D	LPA
1	Dolphin Group	62	159	9	7	3	5	0,515	0,458	0,62	0,83
2	Les Misérables Group	77	254	17	7	5	6	0,556	0,502	0,75	0,92
3	Book Amazon	105	441	27	22	3	5	0,592	0,548	1,14	1,73
4	Email-Eu-core	1005	25571	123	77	15	6	0,623	0,529	18,67	24,59
5	Orkut	1247	15511	207	102	19	8	0,719	0,675	24,23	28,21
6	Political blogosphere	1490	19090	257	87	3	7	0,688	0,557	37,2	30,35
7	Youtube	3088	6695	347	157	23	9	0,794	0,645	67,43	73,28

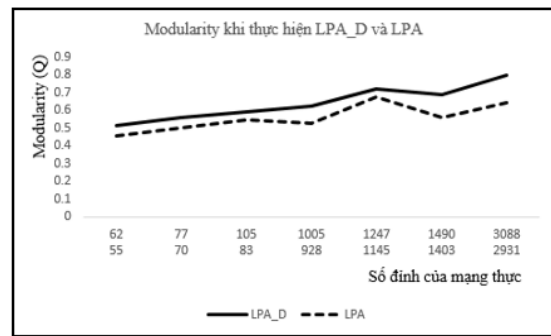


Hình 3. So sánh thời gian thực hiện

Nhận xét: Kết quả đạt được chứng tỏ rằng, thuật toán kết hợp các đỉnh treo tương đương và thuật toán LPA_D do chúng tôi đề xuất là hiệu quả vì đã giảm kích thước của mạng ban đầu thông qua thuật toán rút gọn; đồng thời khi sử dụng hàm gắn nhãn và lan truyền nhãn f_{Degree} đã cho phép việc phân lập cộng đồng nhanh với chất lượng cao, từ đó giảm thời gian tính toán, chất lượng cộng đồng thu được là tốt hơn so với thuật toán LPA [12].

6. Kết luận

Chúng tôi đã giới thiệu một cách tiếp cận mới đối với bài toán phát hiện cộng đồng bao gồm nghiên cứu tính chất của lớp các đỉnh treo tương đương, từ đó đề xuất thuật toán kết hợp các đỉnh treo tương đương để đưa đồ thị mạng ban đầu về đồ thị rút gọn nhằm giảm kích thước của mạng ban đầu. Xuất phát từ thuật toán gắn nhãn LPA, kết hợp thuật toán rút gọn và việc lựa chọn hàm gắn nhãn và lan truyền nhãn f_{Degree} , chúng tôi đã phát triển thuật toán LPA thành thuật toán LPA_D để



Hình 4. So sánh chất lượng cộng đồng

xác định cấu trúc cộng đồng, thuật toán có độ phức tạp tuyến tính. Các kết quả thực nghiệm đã chứng tỏ thuật toán đề xuất có tốc độ chạy nhanh hơn thuật toán LPA, chất lượng cộng đồng thu được là cao hơn.

TÀI LIỆU THAM KHẢO/ REFERENCES

- [1]. G. W. Flake, S. Lawrence, C. L. Giles, and F. M. Coetzee, "Self organization and identification of web communities," *Computer*, vol. 35, no. 3, pp. 66-70, 2002.
- [2]. F. Radicchi, and F. Castellano, "Defining and identifying communities in network," *Proceedings of the National Academy of Sciences of the United States of America*, 2004.
- [3]. M. Girvan, and M. E. Newman, "Community structure in social and biological networks," *Physical review E*, vol. 99, pp. 7821-7826, 2002.
- [4]. J. R. Tyler, and D. M. Wilkinson, "Automated discovery of community structure within organization," *Physical review E*, vol. 15, pp. 723-739, 2003.

- [5]. J. W. Pinney, and D. R. Westhead, *Betweenness-based decomposition methods for social and biological network*. Leeds University Press, 2006.
- [6]. M. J. Rattigan, M. Maier, and D. Jensen, "Graph clustering with network structure indices," *Proceedings of the 24th international conference on Machine learning*, 2007.
- [7]. S. Gregory, *An algorithm to find overlapping community structure in network*. Springer Heidelberg, 2007.
- [8]. S. Fortunato, *Community Detection in Graphs*, Physics Reports, 2010.
- [9]. U. Brandes, "A faster algorithm for betweenness centrality," *Journal of Mathematical sociology*, vol. 25, no. 2, pp. 163-177, 2001.
- [10]. M. E. Newman, and M. Girvan, "Finding and evaluating community structure in networks," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 76, pp. 106-119, 2004.
- [11]. X. Liu, H. M. Cheng, and Z. Y. Zhang, "Evaluation of community detection methods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 9, pp. 1736-1746, 2019.
- [12]. U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale network," *Physics Reports*, vol. 23, pp. 306-323, 2007.
- [13]. Sara E. Garza, and S. E. Schaeffer, "Community detection with the Label Propagation Algorithm," *Elsevier B.V*, vol. 534, pp. 1024-1059, 2019.
- [14]. Z. Wu, A. Efros, and S. X. Yu, "Improving generalization via scalable neighborhood component analysis," In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, Proceedings, Part VII*, 2018, pp. 712-728.
- [15]. Meilian, Zh. Zhang, Zh. Qu, and Yukang, "Overlapping Community Detection Using Label Propagation in Large-Scale Complex Networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, pp. 1736-1749, 2019.
- [16]. A. Zhang, G. Ren, and Y. Lin, "Detecting community structures in network by label propagation with prediction of percolation transition," *The Scientific word Journal*, vol. 14, pp. 315-336, 2014.
- [17]. T. Wu, Y. Guo, and L. Chen, "Integrated structure investigation in complex networks by label propagation," *Physica A: Statistical Mechanics and its Applications*, vol. 448, pp. 68-80, 2016.
- [18]. P. Bonacich, *Introduction to Mathematical Sociology*. Princeton University Press, 2013.
- [19]. L. C. Freeman, "Maintaining the Duality Closeness and Betweenness Centrality," *Social Network*, vol. 44, pp. 153-159, 2016.
- [20]. J. Leskovec, and Krevl, "A. SNAP Datasets Stanford large network dataset collection," 2014. [Online]. Available: <https://snap.stanford.edu>. [Accessed Feb. 26, 2020].