ISSN: 1859-2171
e-ISSN: 2615-9562 TNU Journal of Science and Technology 225(06): 303 - 310

ỨNG DỤNG PHƯƠNG PHÁP PHÂN CỤM PHỔ TRONG BÀI TOÁN PHÁT HIỆN CỘNG ĐỒNG

Nguyễn Hiền Trinh, Vũ Vinh Quang*
Trường Đại học Công nghệ thông tin và Truyền thông - ĐH Thái Nguyên

TÓM TẮT

Ngày nay, phát hiện cộng đồng trên một mạng xã hội đang là hướng nghiên cứu quan trọng trong lĩnh vực khoa học máy tính. Mạng xã hội thường được biểu diễn dưới dạng cấu trúc dữ liệu đồ thị. Chính vì vậy, phát hiện cộng đồng trên mạng xã hội chủ yếu gắn liền với bài toán phân cụm trên đồ thị. Để giải quyết bài toán, đã có rất nhiều thuật toán được quan tâm nghiên cứu. Trong bài báo này, nhóm tác giả sẽ trình bày các kết quả nghiên cứu mới theo hướng tiếp cận sử dụng khái niệm spectrum (phổ) để đưa bài toán phân cụm đồ thị tổng quát về bài toán phân cụm trên véc tơ riêng số thực nhằm giảm số chiều của tập dữ liệu, đồng thời kết hợp kỹ thuật tối ưu hóa hàm Min-cut nhờ sử dụng ma trận Laplace. Hướng tiếp cận này sẽ giảm độ phức tạp tính toán của thuật toán phát hiện cấu trúc cộng đồng trên mạng xã hội. Các kết quả thực nghiệm chạy trên các bộ số liệu thực tế đã khẳng định tính hữu hiệu của thuật toán đề xuất.

Từ khóa: *Khoa học máy tính; mạng xã hội; cấu trúc cộng đồng; khai phá dữ liệu đồ thị; phân cụm đồ thị; phát hiện cộng đồng; phổ.*

Ngày nhận bài: 21/02/2020; Ngày hoàn thiện: 21/5/2020; Ngày đăng: 25/5/2020

THE APPLICATION OF RANGE CLUSTERING METHOD IN COMMUNITY DETECTING PROBLEM

Nguyen Hien Trinh, Vu Vinh Quang*
TNU – University of Information Technology and Communication

ABSTRACT

Nowadays, community detection in graphs has been an important problem in computer science research. Social networks are often expressed in form of structure data graph. Hence, social network community mining mainly deals with graph clustering problem. To solve this problem, many algorithms have been proposed. In this article, the authors present new research results based on the approach of using the concept of spectrum to bring the problem of clustering general graph of clustering problem on vectors of real numbers only, for this reason, the number of dimensions of the data set will be reduced, then we incorporate the techniques of optimizing the Min-cut function using the Laplace matrix. This approach will reduce the calculation complexity and quickly yields the result of social network community structure mining. The effectiveness of proposed algorithm is evidenced by experimental results on real data sets.

Keywords: *Computer science; social network; community structure; graph data mining; graph clustering; community determining, spectrum.*

Received: 21/02/2020; Revised: 21/5/2020; Published: 25/5/2020

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

1. Mở đầu

Phương pháp phân cụm là ứng dụng quan trọng đối với các bài toán đặc trưng cho mô hình đồ thị trong khai phá dữ liệu, nhất là vấn đề xác định các cộng đồng trong mạng xã hội. Phân tích đầu tiên về cấu trúc cộng đồng được thực hiện bởi Weiss và Jacobsen [1] trong việc nghiên cứu tách ra các nhóm làm việc trong một cơ quan chính phủ. Từ đó đến nay đã có nhiều thuật toán được nghiên cứu, phát triển. Có thể kể thêm các tác giả khác như Flake [2], Radicchi [3], [4]… đã đề xuất việc phát hiện cộng đồng theo phương pháp phân cụm đồ thị *G* thành các đồ thị nhỏ hơn với các đặc trưng riêng. Bài toán đặt ra như vậy thuộc lớp NP-Khó, các tác giả Girvan và Newman [5] đã đề xuất phương pháp phân cụm thứ bậc phân chia để phát hiện cộng đồng, ở đó cần tính độ đo trung gian của các cạnh (Betweenness) và từ đó loại trừ cạnh có độ đo trung gian cao nhất. Độ phức tạp của thuật toán này là tương đương $O(k^2n)$ với *k* cạnh cần loại bỏ. Để cải tiến tốc độ của thuật toán Girvan-Newman, đã có nhiều nhóm đề xuất các phương pháp khác nhau như Tyler [6]; Gregory [7]; Brandes [8]. Tuy nhiên, các thuật toán này vẫn có độ phức tạp lớn, khoảng $O(mn^2)$ với *n* là số đỉnh *m* là số cạnh. Hướng cải tiến của nhóm tác giả là tìm cách giảm số chiều của không gian véc tơ (dữ liệu gốc) bằng pháp phân cụm phổ, từ đó rút gọn khối lượng tính toán khi xử lý phát hiện cộng đồng, giảm độ phức tạp tính toán.

Cấu trúc của bài báo gồm 4 phần. Phần 1 (Mở đầu): giới thiệu nội dung nghiên cứu. Phần 2: tóm tắt một số kiến thức cơ bản đề cập đến trong các thuật toán. Phần 3 trình bày bài toán phát hiện cộng đồng bằng phương pháp phân cụm phổ, đề xuất thuật toán. Phần 4 tiến hành thực nghiệm thuật toán trên các bộ dữ liệu.

2. Một số kiến thức cơ bản

2.1. Đồ thị mạng xã hội: Kí hiệu *G* = (*V*, *E*), trong đó *V* là tập các đỉnh biểu diễn các thành viên của mạng xã hội và *E* là tập các cạnh thể hiện mối quan hệ xã hội giữa các thành viên. Một cộng đồng *C* là tập con các đỉnh của *V* sao cho với mỗi đỉnh $v \in C$ có nhiều cạnh kết nối v_i với những đỉnh *u* khác trong *C* và ít cạnh kết nối v_i với những đỉnh *w* khác thuộc $V \backslash C$ [9], [10].

2.2. Đồ thị tương tự: Kí hiệu *G*(*V, E, W*), trong đó $V=\{X_1, X_2,\dots,X_n\}$ là tập các đỉnh, *E* là tập các cạnh $\{(X_i, X_j)\}$ thỏa mãn độ đo $W(X_i, X_j)>0$ trong đó *W* là độ đo tương tự [10], [11].

Đồ thị *G* được phân chia sao cho các cạnh trong nhóm có độ đo tương tự lớn nhất và các cạnh nối các nhóm có độ đo tương tự nhỏ nhất. Để biểu diễn đồ thị *G*, có thể sử dụng các phương pháp:

+ Ma trận kề $W=(w_{i,j})_{n\times n}$ trong đó

$$w_{i,j}=\begin{cases}1,(i,j)\in E,\\0,(i,j)\notin E.\end{cases}$$

+ Ma trận kết nối $A=(a_{i,j})_{n\times n}$ được xác định qua độ đo *W* được định nghĩa theo bài toán.

+ Ma trận bậc $D=(d_{i,j})_{n\times n}$;

$$d_{i,k}=\begin{cases}d(v_i), i=k;\\0, i\neq k.\end{cases}$$ với $d(v_i)$ là bậc của đỉnh v_i.; *G* vô hướng không có trọng số

$$d_{i,j}=\begin{cases}\sum_{k=1}^{n}a_{ij}, i\neq j;\\0, i=j.\end{cases}$$ trong đó a_{ij} là giá trị kết nối giữa các đỉnh; *G* có hướng có trọng số.

Trong nghiên cứu và thực nghiệm, việc xác định sự tương đương giữa 2 đối tượng X_i, X_j được đánh giá theo phân phối Gauss:

$$W(i,j)=\exp\left(-\frac{\|X_i-X_j\|^2}{2\sigma^2}\right),$$

Trong đó σ là độ lệch chuẩn. Trong thực nghiệm, ta lựa chọn giá trị này để điều chỉnh kích thước của cụm, giá trị *W*(*i,j*) càng cao thì liên kết giữa X_i và X_j càng lớn.

Ngoài ra, khoảng cách giữa 2 đối tượng còn được xác định theo khoảng cách Euclid.

$$d(i,j)=\left\|X_i-X_j\right\|.$$

Hiển nhiên mối liên kết cao khi và chỉ khi khoảng cách thấp.

Khi đó, từ khoảng cách có thể xác định được mối liên kết qua nhiều phương pháp khác nhau, một trong những phương pháp được nhóm tác giả lựa chọn là phương pháp phổ [12], [13], nhằm giảm số chiều của dữ liệu được xét và do đó quá trình xác định cộng đồng trên đồ thị tỏ ra hiệu quả hơn rất nhiều về mặt thời gian và độ phức tạp tính toán.

***2.3. Khái niệm phổ*:** Phổ là một tập các giá trị đặc trưng của ma trận *L*: Spec(*L*) = $\begin{pmatrix}\lambda_1 \ldots \lambda_t \\ m_1 \ldots m_t\end{pmatrix}$ trong đó $\lambda_1 \ldots \lambda_t$ là các giá trị khác nhau của giá trị đặc trưng và $m_1 \ldots m_t$ là các hệ số điều chỉnh, *L* là kí hiệu của ma trận Laplace. Ma trận Laplace có một số tính chất cơ bản sau [10], [14], [15].

+ Tổng các phần tử trên các hàng hoặc cột đều bằng không.

+ *L* là ma trận vuông đối xứng, không khả nghịch.

+ *L* là nửa xác định dương.

+ *L* là một toán tử $L: V \rightarrow R$ trong đó V là tập các đỉnh của đồ thị *G*, *R* là tập số thực.

+ Các giá trị riêng của *L* là trực chuẩn (độ lớn bằng 1 và tích vô hướng của 2 véc tơ bằng 0).

+ *L* phụ thuộc vào thứ tự của các đỉnh còn phổ là bất biến đối với đồ thị.

Vấn đề đặt ra là từ ma trận lân cận kề *A* biểu diễn cho đồ thị, ta cần xây dựng nên một ma trận *L* tương đương cũng có khả năng đặc trưng cho đồ thị mà ta đang xét.

Có nhiều cách để xây dựng ma trận *L* từ *A*, ví dụ $L = I - D^{-\frac{1}{2}} \mathrm{A} \mathrm{D}^{-\frac{1}{2}}$ hoặc *L=D-A*.

Từ phương trình $Lu = \lambda u$, ta sẽ xác định được các giá trị riêng và véc tơ riêng của *L*. Các giá trị của một véc tơ riêng chính là các giá trị phổ của đỉnh đồ thị và được sử dụng để tính toán phân cụm đồ thị. Ngoài ra, véc tơ riêng sẽ được chuẩn hóa để dễ dàng tính toán [16], [17]. Như vậy, số chiều của phổ nhỏ hơn số chiều của tập đỉnh ban đầu trong đồ thị vì từ ma trận lân cận kề *A* kích thước $n \times n$, ta đã chuyển về xử lý véc tơ riêng có *n* phần tử. Số cụm cần phân tách trong phương pháp phân cụm phổ tương ứng với giá trị *k* mà chúng ta lựa chọn cho kỹ thuật phân cụm k-mean.

2.4. Tiêu chuẩn đánh giá cộng đồng

Newman và Girvan [18], X. Liu và cộng sự [19] đã đưa ra đại lượng Modularity và tối ưu hóa để đánh giá chất lượng cộng đồng:

$$Q = \frac{1}{2m}\sum_{i,k=1}(a_{\mathrm{ij}} - p_{\mathrm{ij}})\delta(C_i, C_j) \qquad (1)$$

Trong đó *A* là ma trận kề, p_{ik} là số cạnh dự kiến trong *C*, hàm $\delta(C_i, C_j) = 1$ nếu *i, j* thuộc cùng 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:

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

với k_i, k_j là bậc của đỉnh *i* và đỉnh *j*. Nếu gọi n_c là số cộng đồng, l_c là số cạnh nối các đỉnh của cộng đồng *C*, d_c là tổng số bậc của các đỉnh của cộng đồng *C*.

$$Q = \sum_{c=1}^{n_c}(\frac{l_c}{m} - (\frac{d_c}{2m})^2) \qquad (3)$$

Giá trị lớn nhất của *Q* được xác định:

$$Q_{\max} = -\frac{1}{m}\min_C\{[(\mathrm{m}\text{-}\sum_{c=1}^{n_c} l_c) - (m - \sum_{c=1}^{n_c} E_x(l_c))]\} = -\frac{1}{m}\min_C\{\left|Cut_C\right| \text{-} \mathrm{E}_x\mathrm{Cut}_C\} \qquad (4)$$

trong đó $E_x(l_c) = \frac{d_c^2}{4m}$ là số lượng liên kết dự kiến, $|Cut_C| = m - \sum_{c=1}^{n_c} l_c$ là số cạnh liên cộng đồng của *C* và $E_x Cut_C$ là số cạnh dự kiến của các cộng đồng của *C*. Một cộng đồng *C* có Q_{max} đạt giá trị dương và càng lớn thì cộng đồng được xác định càng rõ, tức là việc phân tách cộng đồng là tốt.

3. Phương pháp phân cụm phổ

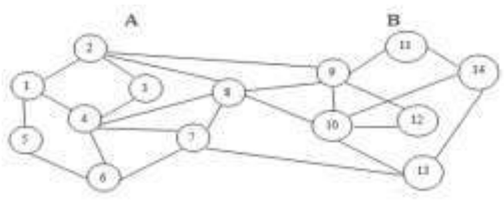
3.1. Bài toán và phương pháp phân cụm phổ

3.1.1. Bài toán: Xét đồ thị mạng *G* = (*V*, *E*) với tập đỉnh $V = \{v_1, ..., v_N\}$, một tập con $Z \subset V$ với độ đo $W(Z_i, Z_j) = \sum_{i \in Z_i, j \in Z_j} A(i,j)$; *A* là ma trận liên kết hoặc ma trận kề. Giá trị của tập con *Z* được xác định $Vol(Z) = \sum_{i \in Z} D_i$, trong đó $D_i = \sum_{j=1}^{N} A(i,j); i = 1..N$ là giá trị của đỉnh thứ *i* của đồ thị. Hãy xác định tập các tập khác rỗng $Z_1, ... Z_k$ sao cho $Z_i \cap Z_j = \phi$ và $Z_1 \cup ... \cup Z_k = V$; đồng thời thỏa mãn tiêu chuẩn phân vùng tốt (trong mỗi nhóm, số lượng cạnh là lớn nhất, nhưng số lượng cạnh giữa 2 nhóm là bé nhất).

3.1.2. Phương pháp phân cụm phổ: Khi mạng được biểu diễn bởi đồ thị, việc phát hiện cộng đồng có mối quan hệ đặc biệt với phân cụm đồ thị. Việc phân chia đồ thị *G* thành 2 nhóm *A*, *B* sao cho trọng số của các cạnh nối các đỉnh từ *A* đến các đỉnh của *B* là nhỏ nhất [20], [21] và các cạnh trong một nhóm có trọng số cao. Sử dụng phương pháp Min-cut với lát cắt $Cut(A,B) = \sum_{i \in A, j \in B} w_{ij}$, w_{ij} là trọng số của cạnh (*i* ,*j*), bài toán được thực hiện với việc chọn lát cắt $Cut(A,B)$ đạt min, còn $Cut(A,A)$ và $Cut(B,B)$ đạt max. Lát cắt Cut được thực hiện theo chuẩn

$$J_{NCut}(A,B) = Cut(A,B)\left(\frac{1}{vol(A)} + \frac{1}{vol(B)}\right)$$

trong đó:



Hình 1. *Minh họa đồ thị phân thành 2 cụm A, B*

$$Vol(A) = \sum_{i \in A} \sum_{j=1}^{n} W_{ij} = \sum_{i \in A} d_i,$$

$$Vol(B) = \sum_{i \in B} \sum_{j=1}^{n} W_{ij} = \sum_{i \in B} d_i$$

Cách tính này có độ phức tạp là $O(|V||E|)$ và sẽ không thực hiện được việc phân chia cụm nếu gặp đỉnh cô lập. Để khắc phục chúng ta sẽ tìm các phương pháp thực hiện khác hiệu quả hơn, một trong các phương pháp đó là sử dụng phương pháp phổ [18], bằng giải pháp dùng véc tơ đặc trưng $X = (v_1, v_2, ..., v_k)$ với *L*= *D*-*A*. hoặc dùng véc tơ đặc trưng $Y = (u_1, u_2, ..., u_k)$ với $L = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$.

Phương pháp của UlrikeVon Luxburg [13] có kết quả phân cụm tốt với thời gian nhanh hơn các phương pháp phân cụm truyền thống.

3.2. Thuật toán đề xuất

Phương pháp nhóm tác giả đề xuất dựa trên nguyên tắc trợ giúp của các véc tơ riêng của ma trận Laplace, thực hiện biến đổi tập hợp các đối tượng dữ liệu ban đầu thành một tập hợp các điểm trong không gian có tọa độ là các phần tử của véc tơ riêng, sau đó các điểm được phân cụm bằng các kỹ thuật tiêu chuẩn k-mean. Khác với các phương pháp khác, nhóm tác giả lựa chọn hàm phân phối Gauss để xác định giá trị ma trận liên kết *A*, đồng thời giá trị *k* được lượng hóa bằng kinh nghiệm sau đó tính và lựa chọn *k* véc tơ đặc trưng từ ma trận *L* và phân cụm k-mean trên tập phổ của tập dữ liệu ban đầu. Quá trình tính toán các giá trị riêng và véc tơ riêng cũng được cải tiến để giảm thời gian tính toán (Thuật toán 3.2.1). Các phương pháp hiện có sử dụng hàm ước lượng riêng ứng với từng bộ dữ liệu để xác định ma trận liên kết *A*. Qua quá trình thực nghiệm với nhiều bộ dữ liệu (https://snap.stanford.edu), nhóm tác giả thấy rằng việc lựa chọn hàm phân phối Gauss là hiệu quả bởi hầu hết các thuộc tính phản ánh đối tượng và tạo ra mối quan hệ tương đồng trong thực tế đều tuân theo phân phối chuẩn Gauss. Phương pháp phổ có thể tính toán nhanh và cho biết khả năng liên kết hay xảy ra cũng như khả năng tốt nhất khi thực hiện kết nạp vào cộng đồng, điều này được đánh giá qua giá trị modularity xác định chất lượng cộng đồng.

Mô tả các thuật toán:

Thuật toán 3.2.1 (Xác định vector riêng của ma trận V)

Input : Ma trận *V* có số chiều $n \times n$; Output: Giá trị véc tơ riêng

Bước 1. Khởi động $x = u^{(0)}; \lambda = \lambda^{(0)}$. Gán $z^{(0)} = (x * x)^{-\frac{1}{2}} x$ sao cho $\left\| z^{(0)} \right\| = 1$. Đặt *t=1*.

Bước 2.

+ Xác định ma trận $M = V - \lambda I$;

+ Giải phương trình $My^{(t)} = z^{(t-1)}$; tính $z^{(t)} = (y^{(t)^*} y^{(t)})^{-\frac{1}{2}} y^{(t)}$;

Bước 3.

+ Tính $\rho_t = \dfrac{z_{k_h}^{(t)}}{y_{k_h}^{(t)}}$; Nếu *V* là ma trận Hermitian thì tính $\mu_t = (z^{(t-1)^*} y^{(t)})^{-1}$

+ Xác định $\lambda^{(t)} = \lambda^{(t-1)} + \rho_{t^*}$ hoặc $\lambda^{(t)} = \lambda^{(t-1)} + \mu_{t^*}$

Bước 4: *t:=t+1*; Quay lại Bước 2. Điều kiện dừng lặp $\left\| z^{(t)} - z^{(t-1)} \right\| \le \varepsilon$

Thuật toán 3.2.2 (Thuật toán SC_NT)

Input: Cho tập dữ liệu $P \in R^{NxF}$, *N*: số điểm dữ liệu, *F* số chiều không gian, *k* số cộng đồng.

Output: Các cộng đồng $Z_1, Z_2, ..., Z_k$ với $Z_i = \{i | y_i \in C_i, i=1..k\}$

1. For $P_i \in P$ $(i \in 1..N)$ *if* $(conect_{P_j \in P}(P_i, P_j)$ *then*

$A(i,j) = \exp(-\dfrac{\left\| p_i - p_j \right\|}{2\sigma^2})$; $i, j \in 1..N$

2. For $P_i \in P$ *deg(*v_i*)=deg(*P_i*);* $P_i \in P$

3. For $i, j \in 1..N$ $D_{ij} = diag(v_1, v_2, ..., v_N)$

4. Comput L=D-A // Ma trận Laplace

5. Comput k eigen-vector $u_1, u_2, ..., u_k$ *của L sao cho* $Lu = \lambda u$

6. Select $u \in \{u_1, u_2, ..., u_k\}$ *// chọn u từ tập các vecto* $u_1, u_2, ..., u_k$ *7. Select* $y \in s\tan d\{u\}$ *// chọn các điểm* $y_i \in R^k$ *từ tập các vecto u, chuẩn hóa u được y*

8. $C_i = k\_means(y)$*, i=1..k//Phân cụm các điểm* y_i *vào k cụm* $C_1, C_2, ..., C_k$ *theo k-Means*

9. Return $Z_i = \{P_i | y_i \in C_j\}$

10. Comput Q

Nhận xét:

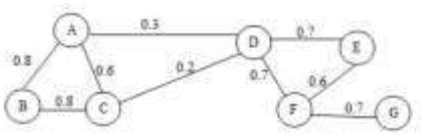
+ Khi sử dụng hàm phân phối Gauss xác định ma trận liên kết *A* thì chất lượng các cộng đồng tốt hơn, mối liên kết giữa 2 đối tượng x_i, x_j là cao nếu các đối tượng này rất giống nhau, sự giống nhau (tương đồng) giữa 2 đối tượng x_i, x_j được xác định theo phân phối Gauss.

+ Do số các véc tơ riêng trong *y* là nhỏ hơn nhiều số chiều không gian *F* nên hiển nhiên việc thực hiện thuật toán k_mean trên tập *y* sẽ giảm độ phức tạp tính toán trên toàn không gian.

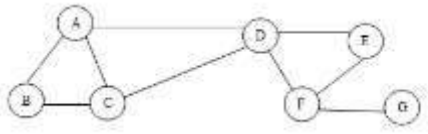
+ Vì số cộng đồng cần xác định là *k* và từ thuật toán k-Means (bước 8) nên dễ thấy độ phức tạp của thuật toán được đánh giá tương đương với $O(k * N^2)$.

4. Các kết quả thực nghiệm

Để kiểm tra độ chính xác của thuật toán, nhóm tác giả tiến hành thử nghiệm thuật toán với một số bộ dữ liệu cụ thể trong đó các bộ dữ liệu test (số liệu nhỏ) kiểm tra độ chính xác của thuật toán khi phân cụm. Các bộ dữ liệu thực lớn [22] nhằm so sánh thuật toán đề xuất với các thuật toán khác. Thực nghiệm được thực hiện trên môi trường Matlab version 7.0.



Hình 2. *Mạng 7 đỉnh và các liên kết*



Hình 3. *Mạng và trọng số liên kết*

***4.1. Ví dụ 1*:** Xét đồ thị mạng gồm 7 đối tượng A(6,8,9); B(8,8,8); C(9,8,7); D(2,3,1); E(1,3,4); F(4,2,3); G(2,3,4) và các liên kết như đồ thị tương ứng biểu diễn mạng (hình 2):

Tiến hành thực hiện thuật toán theo các bước sau

Bước 1, 2, 3: Giá trị của ma trận liên kết *A* (xác định theo phân phối Gauss) và ma trận bậc *D*:

$$A=\begin{bmatrix} 0 & 0.8 & 0.6 & 0.3 & 0 & 0 & 0 \\ 0.8 & 0 & 0.8 & 0 & 0 & 0 & 0 \\ 0.6 & 0.8 & 0 & 0.2 & 0 & 0 & 0 \\ 0.3 & 0 & 0.2 & 0 & 0.7 & 0.7 & 0 \\ 0 & 0 & 0 & 0.7 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0.7 & 0.6 & 0 & 0.7 \\ 0 & 0 & 0 & 0 & 0 & 0.7 & 0 \end{bmatrix} \quad D=\begin{bmatrix} 1.7 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.7 \end{bmatrix} \quad L=\begin{bmatrix} 1.7 & -0.8 & -0.6 & -0.3 & 0 & 0 & 0 \\ -0.8 & 1.6 & -0.8 & 0 & 0 & 0 & 0 \\ -0.6 & -0.8 & 1.6 & -0.2 & 0 & 0 & 0 \\ -0.3 & 0 & -0.2 & 1.9 & -0.7 & 0.7 & 0 \\ 0 & 0 & 0 & -0.7 & 1.3 & -0.6 & 0 \\ 0 & 0 & 0 & -0.7 & -0.6 & 2.0 & -0.7 \\ 0 & 0 & 0 & 0 & 0 & -0.7 & 0.7 \end{bmatrix}$$

Bước 4: Ma trận *L* được xác định: *L=D-A*

Bước 5: Họ véc tơ đặc trưng *x* được xác định $Lx = \lambda x$; Giá trị riêng λ

Bước 6, 7: Chọn $U \in R^3$; *k*= 3; Chuẩn hóa *U* ta có *Y*:

```
Y =

    0.6944   -0.7130   -0.1040
    0.6166   -0.7546   -0.2244
    0.6663   -0.7301   -0.1516
    0.6957    0.2613    0.6691
    0.5140    0.3830    0.7666
    0.7316    0.6671    0.1403
    0.3973    0.5213   -0.7553
```

```
U =

    0.3780   -0.3875   -0.0566
    0.3780   -0.4626   -0.1376
    0.3780   -0.4142   -0.0860
    0.3780    0.1420    0.3635
    0.3780    0.2818    0.5628
    0.3780    0.3446    0.0725
    0.3780    0.4960   -0.7186
```

$$\lambda=\begin{bmatrix} 0 \\ 0.2136 \\ 0.7706 \\ 2.1882 \\ 2.2588 \\ 2.5136 \\ 2.8551 \end{bmatrix}$$

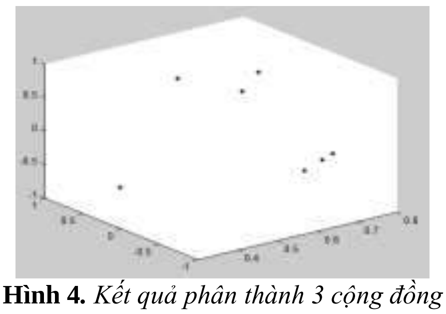
```
X =

    0.3780   -0.3875   -0.0566    0.4486    0.4929   -0.4357    0.2641
    0.3780   -0.4626   -0.1376   -0.3018    0.2895    0.6212   -0.2519
    0.3780   -0.4142   -0.0860   -0.2267   -0.7313   -0.2737    0.1311
    0.3780    0.1420    0.3635    0.5281   -0.2275    0.0724   -0.6075
    0.3780    0.2818    0.5628   -0.5645    0.2526   -0.2796    0.0192
    0.3780    0.3446    0.0725    0.2196   -0.1383    0.4811    0.6591
    0.3780    0.4960   -0.7186   -0.1033    0.0621   -0.1857   -0.2141
```

Bước 8, 9: Phân cụm các điểm $(Y_i), i = 1..7$ vào 3 cụm $1, 2, 3$ bằng thuật toán k-Means:

Bảng 1. *Kết quả phân cụm trong ví dụ 1*

Đối tượng	cộng đồng
A	1
B	1
C	1
D	2
E	2
F	2
G	3



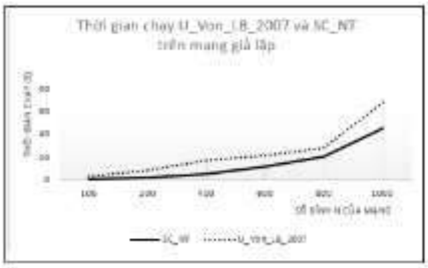
Hình 4. *Kết quả phân thành 3 cộng đồng*

Nhận xét: Thuật toán đề xuất thực hiện phân cụm là tốt, các cộng đồng thu được là hợp lý.

4.2. Ví dụ 2. Xét bộ dữ liệu giả định với kích thước lớn, tiến hành thực hiện thuật toán SC_NT và so sánh với thuật toán của UlrikeVon Luxburg [13]. Nhóm tác giả thu được kết quả trong bảng 2 (n – số đỉnh, m-số cạnh, k-số cộng đồng, t-thời gian).

Bảng 2. *Kết quả so sánh 2 thuật toán (Đơn vị tính thời gian chạy:giây)*

Stt	n	m	k	t_U_Von_LB	t_SC_NT
1	100	500	10	2,45	0,65
2	200	1000	25	7,9	1,85
3	400	2000	32	16,8	4,75
4	600	3000	43	21,3	11,8
5	800	4000	50	28,16	20,8
6	1000	5000	67	68,6	45,2

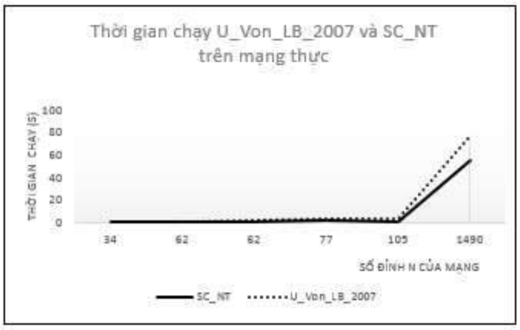


Hình 5. *So sánh thời gian chạy*

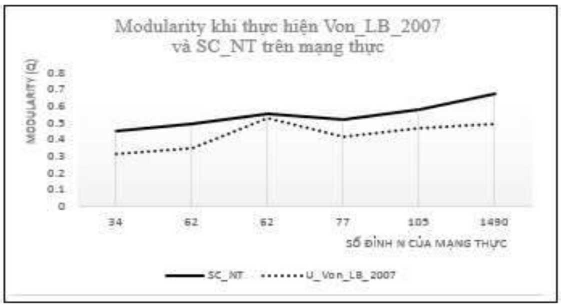
4.3. Ví dụ 3. Nhóm tác giả xét các bộ dữ liệu thực chuẩn [22] trên thế giới, thử nghiệm thuật toán, tính toán chất lượng các cộng đồng thu được đồng thời so sánh với thuật toán của UlrikeVon Luxburg [13]. Kết quả thu được thể hiện ở bảng 3 **(**n-số đỉnh, m-số cạnh, k-số cộng đồng, Q-chất lượng cộng đồng, t-thời gian).

Bảng 3. *Kết quả so sánh 2 thuật toán (Đơn vị tính thời gian chạy: giây)*

Stt	Bộ dữ liệu	n	m	k	Q/t	
					U_Von_LB_2007	SC_NT
1	*Karate Club*	34	78	2	0,31/1,35	0,45/0,53
2	*Dolphin Group*	62	159	3	0,35/1,55	0,49/0,86
3				2	0,53/2,05	0,55/0,97
4	*Les Misérables Group*	77	254	5	0,42/3,74	0,52/1,93
5	*Book Amazon*	105	441	3	0,47/3,93	0,58/1,26
6	*Political blogosphere*	1490	19090	3	0,49/77,35	0,67/55.23



Hình 6. *So sánh thời gian chạy*



Hình 7. *So sánh chất lượng cộng đồng*

Nhận xét. Qua các kết quả thực nghiệm trên các bộ dữ liệu với kích thước lớn được đưa ra trong bảng 3 và các hình 6 - 7, chúng ta thấy rằng thuật toán đề xuất đã thực hiện xác định các cộng đồng là tốt, thời gian chạy của thuật toán đề xuất là nhanh hơn (trung bình khoảng 30%) so với thuật toán của UlrikeVon Luxburg [13]. Chất lượng các cộng đồng thu được cũng cao hơn (trung bình khoảng 23%). Điều này khẳng định tính hiệu quả của thuật toán đề xuất.

5. Kết luận

Nhóm tác giả đã giới thiệu một phương pháp tổng hợp tiếp cận theo hướng giảm số chiều của dữ liệu (dạng ma trận, có thể đa chiều) xuống chỉ còn ở dạng véc tơ (chuỗi số thực), ngoài ra còn phối hợp với ý tưởng tối ưu hóa hàm Min-cut nhờ sử dụng ma trận Laplace, do đó rất hiệu quả cho quá trình xử lý phát hiện cấu trúc cộng đồng trên mạng xã hội. Nền tảng của phương pháp đề xuất là kỹ thuật Spectrum (phân cụm phổ). Phương pháp này có thể đảm bảo lựa chọn được số lượng cộng đồng hợp lý cho mạng. Nhóm tác giả đã chứng minh thuật toán đề xuất có hiệu quả trong việc phát hiện cộng đồng, đồng thời cho thấy phương pháp đề xuất có thể sử dụng để làm sáng tỏ cấu trúc phức tạp của các hệ thống mạng xã hội (mà thực tế cần phải mô tả bằng không gian đa chiều). Thời gian thực hiện thuật toán là nhanh hơn so với thuật toán của UlrikeVon Luxburg. Chất lượng các cộng đồng thu được cũng tốt hơn. Trong thời gian tiếp theo, nhóm tác giả sẽ tiếp tục nghiên cứu mở rộng, hoàn thiện các thuật toán (cho đồ thị có hướng, ma trận thực, đa chiều…) và cải tiến các kỹ thuật phân cụm đồ thị nhằm phát hiện nhanh các cộng đồng có chất lượng cao phục vụ cho việc phân tích và khai thác thông tin trên mạng xã hội.

TÀI LIỆU THAM KHẢO/ REFERENCES

[1]. R. S. Weiss, and E. Jacobsen, “A Method for the analysis of the strucre of complex organizations,” *American Sociological review*, vol. 20 , pp. 661-668, 1999.

[2]. G. W. Flake, and W. Lawrence, “Efficient identìication of web communities,” In Proceedings of the sixth ACM SIGKDD, 2000.

[3]. 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.

[4]. F. Radicchi, and S. Fortunato, “Benchmark graphs for testing community detection algorithms,” *Physical review E.*, vol. 78, pp. 046110, 2008.

[5]. M. Girvan M, and M. E. Newman, “Community structure in social and biological networks,” *Physical review E.*, vol. 99, no. 12, pp. 7821-7826, 2002.

[6]. J. R. Tyler, and D. M. Wilkinson, “Automated discovery of community structure within organization,” *Physical review E*, vol. 15, pp. 723-739, 2003.

[7]. S. Gregory, *An algorithm to find overlapping community structure in network*. Springer Heidelberg, 2007.

[8]. U. Brandes, “A faster algorithm for betweenness centrality,” *Journal of Mathematical sociology*, vol. 2, pp. 163-177, 2007.

[9]. F. Harary, *Graph Theory*. Addison Wesley Reading MA, 1996

[10]. S. Fortunato, “Community Detection in Graphs,” *Physics Reports*, vol. 486, pp. 75-174, 2010

[11]. V. Zografos, and K. Nordberg, “Introduction in Spectral Clustering,” *Physics Reports*, vol. 17, pp. 321-330, 2012

[12]. D. Hamad, *Constrained Spectral embedding for k-way data clusting*. LISIC ULCO, 2014.

[13]. U. von Luxburg, *A Tutorial on Spectral Clustering*. Max Planck Institute for Biological Cybernetics, 2007.

[14]. L. C. Freeman, “A set of measures of centrality based on betweenness,” *Sociometry*, vol. 40, pp. 35-41, 2007.

[15]. M. Clarles, “Spectral Clustering,” *A quick Overview*, vol. 22, pp. 115-124, 2012

[16]. H. Abdi, *The eigenvector-Decomposition*. The University of Texas at Dallas, 2007.

[17]. B. Ruhnau, “Eigenvector-centrality – a node-centrality,” *Social Networks*, vol. 22, pp. 357-365, 2015

[18]. M. E. Newman, and M. Girvan, “Finding and evaluating community structure in networks,” *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 21, pp. 235-251, 2004.

[19]. X. Liu, H. M. Cheng, and Z. Y. Zhang, “Evaluation of community detection methods,” *Physics Reports*, vol. 10, pp. 251-265, 2019.

[20]. S. M. Wagner, “A simple min cut algorithm,” *J.ACM*, vol. 44, pp. 585-591, 2007.

[21]. Wagner, “Between min cut and graph bisection,” *London Springger*, vol. 711, pp. 744-750, 2013.

[22]. J. Leskovec, and Krevl, “A. SNAP Datasets tanford large network dataset collection,” 2014. [Online]. Available: https://snap.stanford.edu. [Accessed Oct. 19, 2019].