Ngân sách tối thiểu phát hiện nguồn thông tin sai lệch trên mạng xã hội trực tuyến, đảm bảo đạt ít nhất một ngưỡng cho trước

Phạm Văn Dũng[1,3], Nguyễn Thị Tuyết Trinh[2], Vũ Chí Quang[3], Hà Thị Hồng Vân[4], Nguyễn Việt Anh[5]
[1] Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội
[2] Học viện Y Dược học cổ truyền Việt Nam, Hà Nội
[3] Học viện An ninh nhân dân, Bộ Công an, Hà Nội
[4] Viện nghiên cứu, phát triển KTNV và kiểm định an ninh thiết bị kỹ thuật ,Cục KTNV, Bộ Công An, Hà Nội
[5] Viện Công nghệ thông tin, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, Hà Nội

Tác giả liên hệ: Phạm Văn Dũng, pvdungc500@gmail.com
Ngày nhận bài: 20/09/2021, ngày sửa chữa: 29/10/2021, ngày duyệt đăng: 15/11/2021
Định danh DOI: 10.32913/mic-ict-research-vn.v2021.n2.1015

Tóm tắt: Phát hiện nguồn phát tán thông tin sai lệch trên mạng xã hội trực tuyến đóng vai trò quan trọng trong việc hạn chế hành vi sai trái trên mạng. Các nghiên cứu gần đây cho thấy, phương pháp đặt máy giám sát có thể phát hiện nguồn thông tin sai lệch. Tuy nhiên, không thể đặt máy giám sát đối với tất cả người dùng mạng vì ngân sách hạn chế.
Trong bài báo này, một mạng xã hội được biểu diễn bởi đồ thị có hướng, mỗi người dùng là một nút trên đồ thị và phát tán thông tin trên đồ thị theo mô hình Bậc độc lập. Trên mô hình này, giả sử biết trước tập nút nghi ngờ sẽ phát tán thông tin sai lệch, chúng tôi đề xuất tìm tập nút nhỏ nhất để đặt giám sát sao cho số nút bị phát hiện đạt ít nhất một ngưỡng cho trước. Ba thuật toán xấp xỉ được đề xuất, bao gồm: Tham lam, phát hiện thông tin sai lệch dựa trên tập mẫu phát hiện và phát hiện thông tin sai lệch dựa trên tập mẫu phát hiện quan trọng. Các thử nghiệm được thực hiện trên bộ dữ liệu của mạng xã hội thực cho thấy các thuật toán của chúng tôi đề xuất vượt trội hơn các thuật toán khác cả về hiệu suất và thời gian thực hiện.

Từ khóa: *Tối ưu hóa, mạng xã hội trực tuyến, phát hiện thông tin sai lệch.*

Title: **Minimum Budget for Misinformation Source Detection in Online Social Networks with Guaranteed to Reach at Least a Given Threshold**

Abstract: Detecting the source of misinformation on social media online plays an important role in curbing online misconduct. Recent studies show that the monitoring method can detect the source of false information. However, it is not possible to set the monitor for all network users because of the limited budget. In this paper, a social network is represented by a directed graph, each user is a node on the graph and propagates information on the graph according to the Degree of Independence model. On this model, assuming that we know in advance the set of suspected nodes will spread false information, we propose to find the smallest set of nodes to set the monitoring so that the number of detected nodes reaches at least a given threshold. Three approximation algorithms are proposed, including: Greedy, Misinformation detection based on detection sample set and Misinformation detection based on important detection sample set. Tests performed on real social network datasets show that our proposed algorithms outperform other base algorithms both in terms of performance and execution time.

Keywords: *Optimization, online social network, misinformation detection.*

I. GIỚI THIỆU

Ngày nay, Mạng xã hội (MXH) đã và đang trở thành một nền tảng quan trọng trong tuyền thông số, có ảnh hưởng không nhỏ đến người dùng. Bên cạnh những lợi ích to lớn thì MXH cũng cho phép phát tán thông tin sai lệch (TTSL) ảnh hưởng đáng kể đến kinh tế và chính trị, v.v.. [1–3]. Do đó, phát hiện TTSL trước khi nó gây ra hậu quả nghiêm trọng là điều hết sức cần thiết. Một số nghiên cứu đã chỉ ra rằng TTSL có thể được phát hiện bằng học máy dựa trên đặc điểm thời gian, cấu trúc, ngôn ngữ, nội dung bài đăng, v.v..[4–6]. Một số khác đề xuất phát hiện TTSL bằng cách

đặt giám sát tại một số nút quan trọng, như phát hiện dịch bệnh [7], TTSL lệch với kích thước tối thiểu [8, 9], phát hiện TTSL bằng tiếp cận heuristic [10], v.v.. Những nghiên cứu về phát hiện thông tin là tiền đề quan trọng để giải quyết bài toán phát hiện và ngăn chặn TTSL trên MXH [11, 11–15].

Mặc dù đã có nhiều nghiên cứu đặt máy giám sát để phát hiện TTSL, tuy nhiên việc phát hiện TTSL trên mạng có hàng trăm nghìn người dùng là không khả thi và không biết có thể đặt bao nhiêu máy giám sát để có thể phát hiện được TTSL và cũng không thể đặt giám sát với tất cả người dùng mạng. Trong bài báo này, chúng tôi nghiên cứu tìm ra tập các nút nhỏ nhất để đặt giám sát sao cho các nút bị phát hiện dự kiến (chức năng phát hiện) đạt ít nhất một ngưỡng cho trước trên mô hình Bậc độc lập IC (Independent Cascade) [16], bài toán được gọi là: Ngân sách tối thiểu phát hiện nguồn thông tin sai lệch MBD (Minimum Budget for Misinformation source Detection). Những đóng góp của bài báo như sau:

- Bài báo chỉ ra rằng MBD là bài toán NP-khó trong trường đạt xấp xỉ $(1-\epsilon)lnn$. Hàm phát hiện có tính chất đơn điệu và submodular và tính toán hàm phát hiện là #P-khó.
- Bài báo đề xuất ba thuật toán xấp xỉ, bao gồm: Thuật toán tham lam Greedy; Phát hiện dựa trên tập mẫu SMD (Sampling based Misinformation Detection) và Phát hiện dựa trên tập mẫu quan trọng ISMD (Important Sampling based Misinformation Detection).

Thực nghiệm được thực hiện trên bộ dữ liệu MXH thực. Kết quả cho thấy các thuật toán được đề xuất mới vượt trội hơn các thuật toán khác cả về hiệu suất và thời gian thực hiện. Bài báo được trình bày bởi 05 phần chính: Phần I - Giới thiệu, Phần II - Mô hình và định nghĩa bài toán, Phần III - Đề xuất thuật toán, Phần IV - Thực nghiệm và đánh giá kết quả, Phần V - Kết luận.

II. MÔ HÌNH VÀ ĐỊNH NGHĨA BÀI TOÁN

Trong phần này, bài báo giới thiệu mô hình IC [16], đây là mô hình được sử dụng phổ biến trong các nghiên cứu bài toán phát tán thông tin trên MXH và định nghĩa bài toán MBD trên mô hình này, các ký hiệu sử dụng trong bài báo được thể hiện trong Bảng I.

1. Mô hình bài toán

Trong bài báo này, chúng tôi sử dụng mô hình IC để mô tả phát tán thông tin trên MXH. Đặc trưng chính của mô hình này là quá trình phát tán thông tin dọc theo các cạnh của đồ thị một cách độc lập với nhau. Trong mô hình IC, mỗi cạnh $(u,v) \in E$ được gán một xác suất ảnh hưởng (Influence Probability) $p(u,v) \in [0,1]$ biểu diễn mức độ ảnh hưởng của nút u với nút v. Nếu $(u,v) \notin E$, thì $p(u,v) = 0$ và $D^t(G,S)$ là tập các nút bị kích hoạt bởi S tại thời điểm t. Quá trình phát tán theo các bước thời gian t rời rạc và kết thúc khi sau mỗi bước không có nút nào được kích hoạt thêm. Nghĩa là, $D^t(G,S) = D^{t-1}(G,S)$.

Bảng I
BẢNG KÝ HIỆU ĐẶC BIỆT

Ký hiệu	Diễn giải
n, m	số nút và số cạnh của đồ thị G
$N_{in}(v), N_{out}(v)$	tập nút vào và ra của v
S	tập nút bi nghi ngờ phát tán TTSL
A	tập nút đặt giám sát
$\mathbb{D}(A)$	hàm ảnh hưởng của tập nút A.
$\hat{\mathbb{D}}(A)$	hàm ước lượng của $\mathbb{D}(A)$
$Cov(A, R_j)$	$= \min\{1, \lvert A \cap R_j \rvert\}$
$Cov_{\mathcal{R}}(A)$	$= \sum_{R_j \in \mathcal{R}} \mathsf{Cov}(A, R_j)$, số tập DS trong $\mathcal{R}$ được bao phủ bởi A
$N_i(\delta, \epsilon)$	$\frac{(2+\frac{2}{3}\epsilon)n}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$, số tập DS tại bước i

- Tại thời điểm $t = 0$, tất cả các nút trong tập nguồn $S = D^0(G,S)$ đều có trạng thái kích hoạt.

- Tại thời điểm $t \geq 1$, mỗi nút $u \in D^{t-1}(G,S)$ có một cơ hội duy nhất kích hoạt đến nút $v \in N_{out}(u)$ với xác suất thành công là $p(u,v)$. Biến cố này có thể được thực hiện bằng cách áp dụng phép thử Bernoulli (Phép tung đồng xu độc lập) với xác suất thành công là $p(u,v)$. Nếu thành công ta thêm v và tập $D^t(G,S)$ và nói rằng u kích hoạt v tại thời điểm t. Nếu nhiều nút kích hoạt v tại thời điểm t, kết quả tương tự xảy ra, v được thêm vào tập $D^t(G,S)$. Một nút ở trạng thái kích hoạt, nó sẽ giữ nguyên trạng thái.

2. Định nghĩa bài toán

Để phát hiện thông tin sai lệch, chúng tôi đề xuất phương pháp tìm ra tập các nút A để đặt máy giám sát, sao cho xác suất phát hiện thông tin sai lệch đạt ngưỡng γ. Gọi tập $S \subseteq V$ là tập các nút bị nghi ngờ phát tán thông tin sai lệch, tức là các nút có khả năng là nguồn gây ra thông tin sai lệch. Mỗi nút được phát hiện là một nguồn phát tán thông tin sai lệch với xác suất $\rho(u) \geq 0$.

Mô hình IC tương đương với mô hình cạnh trực tuyến [16]. Theo đó, chúng ta có thể tạo ra một đồ thị mẫu g từ đồ thị ban đầu G được ký hiệu là $g \sim G$ bằng cách chọn từng cạnh $e = (u,v) \in E$ độc lập, với xác suất chọn cạnh là $p(u,v)$ và không chọn cạnh là $1 - p(u,v)$ Xác suất tạo đồ thị mẫu g từ đồ thị G là:

$$\Pr[g \sim G] = \prod_{e \in E(g)} p(u,v) \prod_{e \in E \setminus E(g)} (1 - p(u,v)) \quad (1)$$

Trong đó, $E(g)$ là tập cạnh của đồ thị mẫu g. Nếu đặt thiết bị giám sát tại nút v, nó sẽ phát hiện TTSL từ các nút được kết nối với nó. Gọi $d_g(u,v)$, là khoảng cách từ u đến

v, thì phải mất $d_g(u,v)$ bước để phát hiện thông tin từ u. Xác suất để tập A phát hiện ra thông tin từ nút u là:

$$\mathbb{D}(A,u) = \sum_{g\sim G} \Pr[g \sim G] R(A,g,u) \quad (2)$$

với:

$$R(A,g,u) = \begin{cases} 1, \text{nếu } d_g(u,A) < \infty, \\ 0, \text{ngược lại} \end{cases} \quad (3)$$

Trong đó, $d_g(u,A) = \min_{v\in A} d(u,v)$, vì xác suất phát hiện nút u là nguồn phát tán thông tin sai lệch là $\rho(u)$, nên xác suất phát hiện của giám sát đặt tại các nút của tập A như sau (hàm mục tiêu):

$$\mathbb{D}(A) = \sum_{u\in S} \rho(u) \sum_{g\sim G} \Pr[g \sim G] R(A,g,u) \quad (4)$$

Bài toán Ngân sách tối thiểu để phát hiện nguồn thông tin sai lệch MBD được định nghĩa như sau:

Định nghĩa 1 (MBD)**:** Một MXH cho bởi đồ thị $G(V,E)$) theo mô hình IC, Tập $S \subseteq V$ là tập các nút được nghi ngờ là nguồn phát tán thông tin sai lệch và mỗi nút $u \in S$ có xác suất $\rho(u) \geq 0$ là nguồn thông tin sai lệch. Cho ngưỡng phát hiện thông tin sai lệch $\gamma > 0$, bài toán đặt ra là tìm tập nhỏ nhất các nút $A \subseteq V$ để đặt giám sát sao cho $\mathbb{D}(A) \geq \gamma$.

Khi các nút có cùng xác suất là nguồn thông tin sai lệch thì giá hàm phát hiện của tập A bằng giá trị ảnh hưởng của A trên đồ thị ngược lại [9]. Vì vậy, tính toán Hàm ảnh hưởng là #P-khó [17], suy ra tính toán Hàm phát hiện $\mathbb{D}(A)$ cũng là #P-khó. Định lý 1 trong [18] đã chứng minh rằng: MBD chỉ có thể đạt xấp xỉ $(1-\epsilon)lnn$ trừ khi $NP \in DTIME(n^{Ologlogn})$.

III. ĐỀ XUẤT THUẬT TOÁN

Trong phần này, bài báo ước tính hàm phát hiện trên mô hình IC và đề xuất các thuật toán xấp xỉ, bao gồm: Greedy, SMD và ISMD cho bài toánMBD.

1. Ước tính giá trị hàm phát hiện (hàm mục tiêu)

Phương pháp ước tính hàm phát hiện $\mathbb{D}(.)$ dựa trên tập mẫu phát hiện DS (Detection Sampling) được thể hiện qua định nghĩa sau:

Định nghĩa 2 (DS)**:** Cho đồ thị $G = (V,E)$ trên mô hình IC, đặt $\rho(S) = \sum_{u\in S} \rho(u)$. Gọi R_j là bộ mẫu phát hiện ngẫu nhiên trên đồ thị G, R_j được tạo như sau:

1) Chọn nút nguồn $u \in V$ với xác suất $\frac{\rho(u)}{\rho(S)}$.
2) Tạo một đồ thị mẫu g từ G, và trả về tập R_j là các nút có thể bắt đầu từ u trong g.

Nút u trong định nghĩa trên được gọi là nguồn của R_j, ký hiệu $src(R_j) = u$. Ω là không gian xác suất của các bộ DS, trong đó xác suất tạo R_j với nút nguồn u (ký hiệu là $R_j(u)$) được tính là:

$$\Pr[R_j(u) \sim \Omega] = \frac{\rho(u)}{\rho(S)} \cdot \sum_{g\sim G: R(R_j,g,u)=1} \Pr[g \sim G] \quad (5)$$

Về cơ bản, vai trò của tập DS tương tự như tập RR (Reachable Reverse) được thiết lập trong việc ước tính hàm phát tán [19–23]. Một biến ngẫu nhiên $X_j(A)$ được định nghĩa như sau:

$$X_j(A) = \begin{cases} 1, \text{ Nếu } R_j \cap A \neq \emptyset \\ 0, \text{ ngược lại} \end{cases} \quad (6)$$

Tương tự bổ đề 2 trong [19], với mọi tập nút $A \in V$, và $A \subseteq V$, gọi $\rho(S) = \sum_{u\in S} \rho(u)$ ta có:

$$\mathbb{D}(A) = \rho(S) \cdot \mathbb{E}[X_j(A)] \quad (7)$$

2. Thuật toán Greedy

Thuật toán Greedy chọn nút u cho vào tập A nếu mức tăng lớn nhất trong việc giảm hiệu suất trong mỗi bước, được định nghĩa như sau:

$$\delta(A,u) = \min(\mathbb{D}(A \cup \{u\}), \gamma) - \mathbb{D}(A) \quad (8)$$

cho đến khi $\mathbb{D}(A) \geq \gamma - \epsilon$, tuy nhiên, không thể áp dụng trực tiếp thuật toán cho MXH vì tính toán hàm phát hiện là #P-khó. Vì vậy, bài báo sử dụng mô phỏng Monte-Carlo (MC) để ước tính hàm phát hiện dựa trên xấp xỉ trung bình mẫu của các lần mô phỏng.

Thuật toán 1: Thuật toán Greedy

Input: Đồ thị $G(V,E)$, tập nguồn $S \subseteq V$, ngưỡng γ
Output: Tập nút đặt giám sát A

```
A ← ∅;
while D(A) < γ − ε do
    u ← arg max_{v∈V\S}(min(D(A ∪ {v}), γ) − D(A));
    A ← A ∪ {u};
end
return A.
```

Theo kết quả chứng minh của bổ đề 2 trong [24], thuật toán Greedy cho tỷ lệ xấp xỉ $(1 - ln\frac{\gamma}{\epsilon})$ với $\epsilon \in (0,\gamma)$. Gọi R là thời gian mô phỏng MC, độ phức tạp của Greedy là $O(Rnk)$, trong đó k là số vòng lặp trong thuật toán, n là số đỉnh của đồ thị.

3. Thuật toán dựa trên tập mẫu phát hiện - SMD

Thuật toán này kết hợp hai kỹ thuật: (1) tạo ra một bộ các DS đủ lớn để ước tính chức năng phát hiện bằng cách áp dụng lý thuyết Martingale [25] và (2) sử dụng thuật toán Greedy để tìm ra tập A đủ tốt đảm bảo chất lượng lời giải.

Ký hiệu $Cov_{\mathcal{R}}(A) = \sum_{R_j \in \mathcal{R}} \min\{1, |A \cap R_j|$ là số bộ DS trong $\mathcal{R}$ được phủ bởi A. Chúng ta thu được ước lượng của $\mathbb{D}(A)$ từ $\mathcal{R}$ như sau:

vì số lượng bộ DS trong $\mathcal{R}$ được bao phủ bởi A. Từ Bổ đề 2 trong [18], chúng ta ước tính $\mathbb{D}(A)$ từ $\mathcal{R}$ như sau:

$$\hat{\mathbb{D}}_{\mathcal{R}}(A) = \frac{\rho(S)}{|\mathcal{R}|}\mathsf{Cov}_{\mathcal{R}}(A) \tag{9}$$

Vì $Cov_R()$ là một hàm đơn điệu và submodular, $\mathbb{D}(.)$ cũng có tính chất tương tự. Tập mẫu DS được tạo ra bằng thuật toán 2. Đầu tiên, chọn một nút nguồn u với xác suất $\frac{\rho(u)}{\rho(S)}$ (dòng 1). Sau đó sử dụng hàng đợi Q để lưu trữ các nút đã truy cập. Trong mỗi bước, thuật toán chọn một nút u trong Q và thêm nó vào R_j. Sau đó, chọn từng nút lân cận v với xác suất $p(u, v)$ theo mô hình cạnh trực tuyến (dòng 8) và đưa vào Q. Quá trình này lặp lại cho đến khi Q rỗng.

Thuật toán 2: Thuật toán tạo mẫu phát hiện DS

Input: Đồ thị $G(V, E)$, tập nguồn $S \subseteq V$
Output: tập nút phát hiện R_j

1. Chọn nút $u \in V$ với xác suất $\Pr[u] = \frac{\rho(u)}{\rho(S)}$;
2. Queue $Q \leftarrow \{u\}$;
3. **while** Q *chưa rỗng* **do**
4. $\quad u \leftarrow Q.pop()$;
5. $\quad R_j \leftarrow R_j \cup \{u\}$;
6. $\quad$ **foreach** $v \in N_{out}(u) \setminus R_j$ **do**
7. $\quad\quad$ **if** $v \notin Q$ **then**
8. $\quad\quad\quad$ Chọn nút v với xác suất $p(u, v)$;
9. $\quad\quad\quad$ **if** *(v được chọn)* **then**
10. $\quad\quad\quad\quad Q.push(v)$;
11. $\quad\quad\quad$ **end**
12. $\quad\quad$ **end**
13. $\quad$ **end**
14. **end**
15. **return** R_j.

Thuật toán SMD được thực hiện như sau: Đầu tiên, thuật toán tạo ra tập $\mathcal{R}$ chứa $N = \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$ tập DS đảm bảo xấp xỉ (δ, ϵ) cho giải pháp tối ưu A^*, nghĩa là:

$$\Pr[(1+\epsilon)\mathbb{D}_{\mathcal{R}}(A^*) \geq \hat{\mathbb{D}}_{\mathcal{R}}(A^*) \geq (1-\epsilon)\mathbb{D}_{\mathcal{R}}(A^*)] \geq 1-\delta \tag{10}$$

bằng cách áp dụng lý thuyết martingale [25]. Trong mỗi vòng lặp chọn nút u có giá trị tăng lớn nhất của hàm phát hiện $\hat{\delta}(A, v)$ như sau:

$$\hat{\delta}(A, v) = \hat{\mathbb{D}}(A \cup \{u\}) - \hat{\mathbb{D}}(A) \tag{11}$$

Nếu giải pháp hiện tại A đạt giá trị $\hat{\mathbb{D}}(A) \geq (\gamma - \epsilon\gamma) - \epsilon$ thì thuật toán trả về A. Mặt khác, kiểm tra số lượng mẫu hiện tại có đủ cho lần lặp tiếp theo không? Nếu số lượng mẫu vẫn đủ thì di chuyển vào các vòng lặp tiếp theo. Nếu số lượng mẫu không đủ, sẽ tạo thêm $(N_i - N)$ bộ mẫu mới (dòng 13) và đặt lại giải pháp hiện tại A, tức là $A \leftarrow \emptyset$ (dòng 15). Thuật toán di chuyển đến bước tiếp theo và chỉ kết thúc khi nó đáp ứng được điều kiện $\hat{\mathbb{D}}(A) \geq (\gamma-\epsilon\gamma)-\epsilon$.

Thuật toán 3: Thuật toán phát hiện *SMD*

Input: $G(V, E)$, $S \subseteq V$, γ,tham số $\epsilon, \delta \in (0, 1)$
Output: Tập nút đặt giám sát A

1. ; $N \leftarrow \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$;
2. Tạo tập $\mathcal{R}$ chứa N tập DS bởi thuật toán. 2
3. $A \leftarrow \emptyset$;
4. **while** *True* **do**
5. $\quad u \leftarrow \arg\max_{v \in V \setminus A} \left(\hat{\mathbb{D}}(A \cup v) - \hat{\mathbb{D}}(A)\right)$
6. $\quad A \leftarrow A \cup \{u\}$;
7. $\quad$ **if** $\hat{\mathbb{D}}(A) \geq (\gamma - \epsilon\gamma) - \epsilon$ **then**
8. $\quad\quad$ **return** A;
9. $\quad$ **else**
10. $\quad\quad i \leftarrow |A| + 1$;
11. $\quad\quad N_i \leftarrow \frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$;
12. $\quad\quad$ **if** $N < N_i$ **then**
13. $\quad\quad\quad$ Tạo thêm $N_i - N$ tập DS đưa vào tập $\mathcal{R}$;
14. $\quad\quad\quad N \leftarrow N_i$;
15. $\quad\quad\quad A \leftarrow \emptyset$;
16. $\quad\quad$ **end**
17. $\quad$ **end**
18. **end**
19. **return** A.

4. Thuật toán dựa trên tập mẫu quan trọng - ISMD

Ý tưởng chính của thuật toán là sử dụng các mẫu phát hiện quan trọng và lý thuyết martingale để ước tính hàm phát hiện. Trong thuật toán 4, để tạo tập IDS, đầu tiên chọn nút nguồn IDS (dòng 1). Sau đó, tính toán xác suất $Pr[E_i], i = 1, \ldots, l(u)$ và chọn một nút ra u_i trong $N_{out}(u)$ với xác suất $\frac{Pr[E_i]}{\phi(u)}$ (dòng 3). Điều này sẽ đảm bảo rằng ít nhất một trong các hàng xóm của u sẽ được kích hoạt. Các nút $v_1, v_2, ..., v_{j-1}$ vẫn không kị kích hoạt và các nút $v_{i+1}, ..., v_l$ sau đó được kích hoạt độc lập với xác suất $p(u, v_j)$ (dòng 7). Phần còn lại của thuật toán này tương tự như Thuật toán 2.

Chi tiết của ISMD được trình bày trong Thuật toán 5. Đầu tiên, ISMD tạo bộ IDS thay vì DS (dòng 2) và sử dụng chúng để ước tính chức năng phát hiện. Thứ hai, trong mỗi trình lặp, SMD cần $\frac{(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$ bộ DS nhưng ISMD chỉ cần $\frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$ tập IDS, với $(q < 1)$. Gọi M là thời gian tạo ra một mẫu, Thuật toán 5 có độ phức tạp thời gian là $O\left(qi_{max}\rho(S)\ln(\binom{n}{i_{max}}/\delta)\epsilon^{-2})M\right)$.

Thuật toán 4: Tạo mẫu phát hiện quan trọng IDS

Input: Đồ thị $G = (V, E)$, tập nguồn $S \subseteq V$
Output: Tập mẫu phát hiện quan trọng R_j;
Chọn nút $u \in V$ với xác suất $\Pr[u] = \frac{\varphi(u)\rho(u)}{\Phi}$;
Tính $\Pr[E_i], i = 1 \ldots l(u)$;
Chọn nút $v_i \in N_{out}(u)$ với xác suất $\frac{\Pr[E_i]}{\varphi(u)}$;
$R_j \leftarrow \{u, v_i\}$;
Queue $Q \leftarrow \{v_i\}$;
for $j = i + 1$ *to* l **do**
  Chọn nút v_j với xác suất $p(u, v_j)$;
  if *(v_j được chọn)* **then**
    $Q.push(v_j)$, $R_j \leftarrow R_j \cup \{v_j\}$;
  end
end
while *Q chưa rỗng* **do**
  $u \leftarrow Q.pop()$;
  foreach $v \in N_{out}(u) \setminus R_j$ **do**
    if $v \notin Q$ **then**
      Chọn nút v với xác suất $p(u, v)$;
      if *(v được chọn)* **then**
        $Q.push(v)$;
        $R_j \leftarrow R_j \cup \{v\}$;
      end
    end
  end
end
return R_j.

Các thuật toán SMD và ISMD cung cấp cùng một kết quả lý thuyết, tuy nhiên, tổng số mẫu ISMD sử dụng thấp hơn SMD và thời gian chạy của ISMD ít hơn thời gian của SMD. Nhận xét này phù hợp với kết quả thực nghiệm trên các tập dữ liệu trong Phần V.

IV. KẾT QUẢ THỰC NGHIỆM

Để đánh giá toàn diện các thuật toán được đề xuất, thực nghiệm tiến hành so sánh các thuật toán đề xuất là Greedy, SMD, ISMD với nhau và so sánh với các thuật toán cơ sở phổ biến là Deegre và Pagerank. Ngoài ra, SMD và ISMD được so sánh với thuật toán OPIM [22], thuật toán lấy mẫu RR mới nhất cho bài toán Tối đa hóa ảnh hưởng. Dữ liệu được lấy từ web: [http://snap.stanford.edu/data/] bao gồm các OSN có quy mô từ hàng nghìn đến hàng triệu cạnh, cụ thể là: Email-Eu-Core [26, 27], Wiki-Vote [28, 29], CA-HepPh [27], Email-Eu-All [27]. Các thuật toán được cài đặt bằng ngôn ngữ Python trên máy tính có cấu hình: CPU Intel Core i7 – 8550U 1,8Ghz, RAM 8GB DDR4 2400MHz, hệ điều hành Linux.

Thuật toán 5: Thuật toán phát hiện *ISMD*

Input: $G(V, E)$, $S \subseteq V$, ngưỡng γ, $\epsilon, \delta \in (0, 1)$
Output: Tập nút đặt giám sát A;
$N \leftarrow \frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(n/\delta)$;
Tạo tập $\mathcal{R}$ chứa N tập mẫu phát hiện IDS;
$A \leftarrow \emptyset$;
while *True* **do**
  $u \leftarrow \arg\max_{v \in V \setminus A} \left(\hat{\mathbb{D}}(A \cup v) - \hat{\mathbb{D}}(A)\right)$;
  $A \leftarrow A \cup \{u\}$;
  if $\hat{\mathbb{D}}(A) \geq \gamma - \epsilon\gamma - \epsilon$ **then**
    return A;
  else
  end
  $i \leftarrow |A| + 1$;
  $N_i \leftarrow \frac{q(2+\frac{2}{3}\epsilon)\rho(S)}{\epsilon^2(\gamma-\epsilon\gamma)} \ln(\binom{n}{i}/\delta)$;
  if $N < N_i$ **then**
    Tạo thêm $N_i - N$ tập IDS thêm vào tập $\mathcal{R}$;
    $N \leftarrow N_i$;
    $A \leftarrow \emptyset$;
  end
end
return A.

Bảng II
DỮ LIỆU THỰC NGHIỆM

Bộ dữ liệu	Nút	Cạnh	Kiểu	Bậc TB
Email-Eu-Core	1,005	25,571	Có hướng	25,44
Wiki-Vote	7,115	103,689	Có hướng	14,57
CA-HepPh	12,008	118,521	Vô hướng	9,87
Email-Eu-All	265,214	420,045	Có hướng	1,58

1. Cài đặt thực nghiệm

Thực nghiệm dựa trên mô hình Trivalency [6, 16, 23, 30] để chọn trọng số của các cạnh. Xác suất ảnh hưởng được chon ngẫu nhiên từ tập được xác định trước, trong thực nghiệm này, ta chọn $p(u, v) \in \{0,001, 0,01, 0,1\}$. Ý tưởng của mô hình này là các cạnh có trọng số $0,001$ thì nút đầu được coi là có mức độ ảnh hưởng thấp, $0,01$ tương ứng với mức ảnh hưởng trung bình và $0,1$ ảnh hưởng mức cao. Tham số đầu vào được chọn như sau: $\delta = 1/n$ theo các nghiên cứu [20–23]. Các nút nghi ngờ được chọn ngẫu nhiên với kích thước $n/2$ và xác suất $\rho(u)$ được chọn ngẫu nhiên trong $[0, 1]$. Ngưỡng γ và tham số ϵ được chọn tùy thuộc vào quy mô của mạng. Ký hiệu $\Psi = \gamma/\rho(S)$ phản ánh mối quan hệ giữa γ và $\rho(S)$. Giá trị của các tham số này được mô tả trong Bảng III. Trong thuật toán cơ sở, phương pháp Monte Carlo được sử dụng 10.000 lần để ước tính hàm phát hiện. Đối với mỗi thuật toán, chạy 10 lần để lấy kết quả trung bình. Thời gian chạy của mỗi thuật toán

được giới hạn trong vòng 24 giờ.

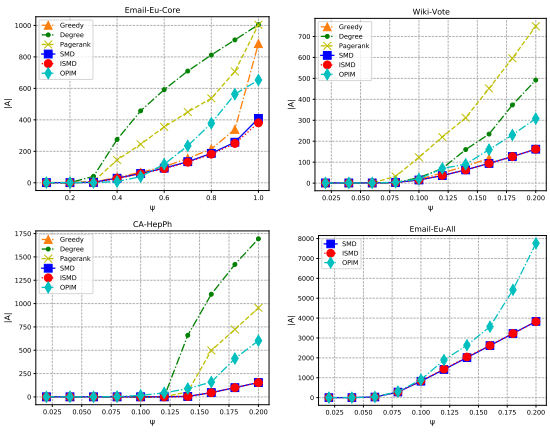
Bảng III
THAM SỐ ĐẦU VÀO CHO CÁC MẠNG

Bộ dữ liệu	$\rho(S)$	Ψ	ϵ
Email-Eu-Core	249,33	1,0	0,01
Wiki-Vote	1784,78	0,2	0,01
CA-HepPh	3009,29	0,2	0,01
Email-Eu-All	171217	0,1	0,1

2. Đánh giá kết quả

a) So sánh hiệu suất thuật toán

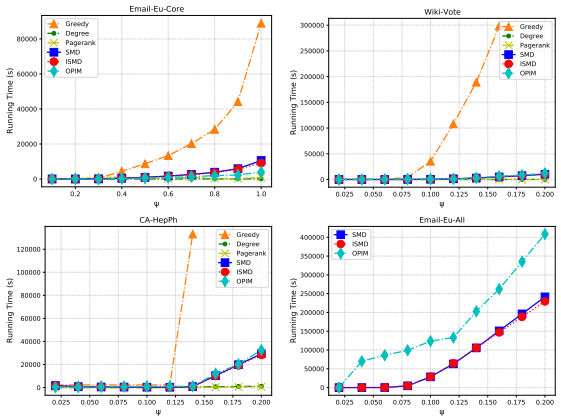
Hiệu suất của các thuật toán được xác định bằng kích thước của tập nút đặt giám sát. Thuật toán tốt hơn trả về tập nút đặt giám sát kích thước nhỏ hơn. Hình 1 cho thấy SMD và ISMD có cùng hiệu suất vượt trội hơn nhiều các thuật toán khác trong mọi trường hợp, giá trị của Ψ càng lớn thì khoảng cách vượt càng lớn. Cụ thể, với cùng giá trị Ψ, SMD và ISMD tốt hơn OPIM tới 3,9 lần, tốt hơn Greedy 2,3 lần. Các thuật toán tác giả đề xuất cũng tốt hơn nhiều lần so với các thuật toán cơ sở là Degree và Pagerank. Điều này chứng tỏ rằng thuật toán SMD và ISMD có hiệu suất tốt hơn các thuật toán khác. Ngoài ra, việc ước lượng giá trị hàm phát hiện bằng các bộ mẫu DS và IDS cho kết quả tốt hơn so với mô phỏng MC trong thuật toán Greedy.



Hình 1. So sánh hiệu suất thuật toán

b) So sánh thời gian chạy thuật toán

Trong thực nghiệm này, thời gian được tính bằng giây (s). Hình 2 cho thấy, thời gian chạy của thuật toán SMD và ISMD đều vượt trội hơn đáng kể so với các thuật toán còn lại. Trong đó SMD, ISMD nhanh hơn OPIM tới 1,5 lần trên hầu hết các mạng, OPIM chỉ cho thời gian tốt hơn trên mạng Email-Eu-Core. Điều này là do OPIM mất nhiều thời gian để tìm kiếm nhị phân cho các giải pháp. So với Greedy, SMD nhanh hơn Greedy tới 10,2 lần và ISMD nhanh hơn Greedy tới 12,4 lần. Đối với các mạng lớn như Email-Eu All Greedy không thể hoàn thành trong thời gian giới hạn (24 giờ) trong khi các thuật toán SMD và ISMD vẫn hoạt động và cho kết quả tốt. Điều này cho thấy việc ước lượng hàm phát hiện bằng DS và IDS nhanh hơn so với việc sử dụng phương pháp mô phỏng Monte Carlo truyền thống của Greedy. So sánh riêng SMD và ISMD thì thời gian chạy trung bình của ISMD nhanh hơn SMD tới 1,4, nguyên nhân chính là do số lượng mẫu yêu cầu của ISMD thấp hơn so với SMD. Các thuật toán cơ sở có thời gian chạy nhỏ nhất vì chúng là các thuật toán heuristic đơn giản với độ phức tạp thấp.



Hình 2. So sánh thời gian chạy thuật toán

V. KẾT LUẬN

Trong bài báo này, bài toán Ngân sách tối thiểu phát hiện nguồn thông tin sai lệch MBD được nghiên cứu nhằm mục đích tìm ra tập hợp các nút nhỏ nhất để đặt giám sát sao cho phát hiện được thông tin sai lệch từ các nút bị nghi ngờ, chức năng phát hiện dự kiến đảm bảo đạt ít nhất một ngưỡng cho trước $\gamma > 0$. Trên mô hình IC, bài báo đề xuất 03 thuật toán xấp xỉ để giải quyết bài toán MB, bao gồm: Greedy, SMD và ISMD. Thực nghiệm cho thấy thuật toán SMD và ISMD vượt trội hơn các thuật toán khác. Tuy nhiên, thời gian thực hiện thuật toán chưa thật sự tốt để áp dụng cho các mạng có quy mô hàng triệu đỉnh và cạnh. Thời gian tới, chúng tôi nghiên cứu cải thiện thời gian chạy của các thuật toán để có thể áp dụng cho các mạng lớn hơn.

LỜI CẢM ƠN

Công trình này được hỗ trợ bởi Viện Hàn lâm Khoa học và Công nghệ Việt Nam, mã đề tài: VAST01.05 / 21-22.

TÀI LIỆU THAM KHẢO

[1] "Misinformation on social media led to pune violence: Minister," in *https://www.ndtv.com/mumbai-news/misinformation-on-social-media-led-to-pune-violence-minister-1795562*, 2018.

[2] P. Domm, "False rumor of explosion at white house causes stocks to briefly plunge; ap confirms its twitter feed was hacked," in *Available: https://www.cnbc.com/id/100646197*, 2013.

[3] V. Luckerson, "Fear, misinformation, and social media complicate ebola fight," in *http://time.com/3479254/ebola-social-media/*, 2014.

[4] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang, "Prominent features of rumor propagation in online social media," in *2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, December 7-10, 2013*, 2013, pp. 1103–1108. [Online]. Available: https://doi.org/10.1109/ICDM.2013.61

[5] J. Ma, W. Gao, and K. Wong, "Detect rumors in microblog posts using propagation structure via kernel learning," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, 2017, pp. 708–717. [Online]. Available: https://doi.org/10.18653/v1/P17-1066

[6] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, 2010, pp. 88–97. [Online]. Available: https://doi.org/10.1109/ICDM.2010.118

[7] J. Leskovec, M. McGlohon, C. Faloutsos, N. S. Glance, and M. Hurst, "Patterns of cascading behavior in large blog graphs," in *Proceedings of the Seventh SIAM International Conference on Data Mining, April 26-28, 2007, Minneapolis, Minnesota, USA*, 2007, pp. 551–556. [Online]. Available: https://doi.org/10.1137/1.9781611972771.60

[8] H. Zhang, A. Kuhnle, H. Zhang, and M. T. Thai, "Detecting misinformation in online social networks before it is too late," in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016, San Francisco, CA, USA, August 18-21, 2016*, 2016, pp. 541–548. [Online]. Available: https://doi.org/10.1109/ASONAM.2016.7752288

[9] H. Zhang, H. Zhang, X. Li, and M. T. Thai, "Limiting the spread of misinformation while effectively raising awareness in social networks," in *Computational Social Networks - 4th International Conference, CSoNet 2015, Beijing, China, August 4-6, 2015, Proceedings*, 2015, pp. 35–47. [Online]. Available: https://doi.org/10.1007/978-3-319-21786-4_4

[10] H. Zhang, M. A. Alim, X. Li, M. T. Thai, and H. T. Nguyen, "Misinformation in online social networks: Detect them all with a limited budget," *ACM Trans. Inf. Syst.*, vol. 34, no. 3, pp. 18:1–18:24, 2016. [Online]. Available: http://doi.acm.org/10.1145/2885494

[11] C. V. Pham, H. M. Dinh, H. D. Nguyen, H. T. Dang, and H. X. Hoang, "Limiting the spread of epidemics within time constraint on online social networks," in *Proceedings of the Eighth International Symposium on Information and Communication Technology, Nha Trang City, Viet Nam, December 7-8, 2017*, 2017, pp. 262–269. [Online]. Available: https://doi.org/10.1145/3155133.3155157

[12] D. V. Pham, G. L. Nguyen, T. N. Nguyen, C. V. Pham, and A. V. Nguyen, "Multi-topic misinformation blocking with budget constraint on online social networks," *IEEE Access*, vol. 8, pp. 78 879–78 889, 2020.

[13] E. B. Khalil, B. N. Dilkina, and L. Song, "Scalable diffusion-aware optimization of network topology," in *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, 2014, pp. 1226–1235. [Online]. Available: http://doi.acm.org/10.1145/2623330.2623704

[14] C. V. Pham, M. T. Thai, H. V. Duong, B. Q. Bui, and H. X. Hoang, "Maximizing misinformation restriction within time and budget constraints," *J. Comb. Optim.*, vol. 35, no. 4, pp. 1202–1240, 2018. [Online]. Available: https://doi.org/10.1007/s10878-018-0252-3

[15] H. T. Nguyen, A. Cano, V. Tam, and T. N. Dinh, "Blocking self-avoiding walks stops cyber-epidemics: A scalable gpu-based approach," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2019.

[16] D. Kempe, J. M. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003*, 2003, pp. 137–146. [Online]. Available: http://doi.acm.org/10.1145/956750.956769

[17] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincón, X. Sun, Y. Wang, W. Wei, and Y. Yuan, "Influence maximization in social networks when negative opinions may emerge and propagate," in *Proceedings of the Eleventh SIAM International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona, USA*, 2011, pp. 379–390. [Online]. Available: https://doi.org/10.1137/1.9781611972818.33

[18] C. V. Pham, D. V. Pham, B. Q. Bui, and A. V. Nguyen, "Minimum budget for misinformation detection in online social networks with provable guarantees," *Optimization Letters*, pp. 1–30, 2021.

[19] C. Borgs, M. Brautbar, J. T. Chayes, and B. Lucier, "Maximizing social influence in nearly optimal time," in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, 2014, pp. 946–957. [Online]. Available: https://doi.org/10.1137/1.9781611973402.70

[20] Y. Tang, X. Xiao, and Y. Shi, "Influence maximization: near-optimal time complexity meets practical efficiency," in *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, 2014, pp. 75–86. [Online]. Available: http://doi.acm.org/10.1145/2588555.2593670

[21] Y. Tang, Y. Shi, and X. Xiao, "Influence maximization in near-linear time: A martingale approach," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, 2015, pp. 1539–1554. [Online]. Available: http://doi.acm.org/10.1145/2723372.2723734

[22] G. Das, C. M. Jermaine, and P. A. Bernstein, Eds., *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*. ACM, 2018. [Online]. Available: https://doi.org/10.1145/3183713

[23] H. T. Nguyen, M. T. Thai, and T. N. Dinh, "Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks," in *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, 2016, pp. 695–710. [Online]. Available: http://doi.acm.org/10.1145/2882903.2915207

[24] A. Goyal, F. Bonchi, L. V. S. Lakshmanan, and S. Venkatasubramanian, "On minimizing budget and time in influence propagation over social networks," *Social Netw. Analys. Mining*, vol. 3, no. 2, pp. 179–192, 2013. [Online]. Available: https://doi.org/10.1007/s13278-012-0062-z

[25] F. R. K. Chung and L. Lu, "Survey: Concentration inequalities and martingale inequalities: A survey," *Internet Mathematics*, vol. 3, no. 1, pp. 79–127, 2006. [Online]. Available: https://doi.org/10.1080/15427951.2006.10129115

[26] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich,

"Local higher-order graph clustering," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, 2017, pp. 555–564. [Online]. Available: https://doi.org/10.1145/3097983.3098069

[27] J. Leskovec, J. M. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *TKDD*, vol. 1, no. 1, p. 2, 2007. [Online]. Available: https://doi.org/10.1145/1217299.1217301

[28] J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg, "Signed networks in social media," in *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, Atlanta, Georgia, USA, April 10-15, 2010*, 2010, pp. 1361–1370. [Online]. Available: https://doi.org/10.1145/1753326.1753532

[29] ——, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, 2010, pp. 641–650. [Online]. Available: https://doi.org/10.1145/1772690.1772756

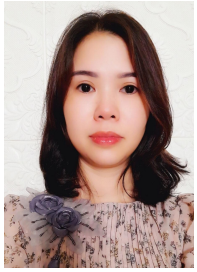
[30] Y. Zhang and B. A. Prakash, "Data-aware vaccine allocation over large networks," *TKDD*, vol. 10, no. 2, pp. 20:1–20:32, 2015. [Online]. Available: http://doi.acm.org/10.1145/2803176

SƠ LƯỢC VỀ TÁC GIẢ

Phạm Văn Dũng

Nhận bằng thạc sĩ Công nghệ thông tin năm 2012 tại Học viện Kỹ thuật Quân sự. Đang làm nghiên cứu sinh tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.
Hiện là giảng viên Khoa An ninh thông tin, Học viện An ninh nhân dân.
Lĩnh vực nghiên cứu: Tối ưu, phân tích mạng xã hội, an ninh mạng và phòng chống tội phạm sử dụng công nghệ cao.
Email: pvdungc500@gmail.com

Nguyễn Thị Tuyết Trinh



Nhận bằng thạc sĩ Công nghệ thông tin năm 2012 tại Học viện Kỹ thuật quân sự.
Hiện là giảng viên bộ môn Toán tin, chuyên viên phòng Công nghệ thông tin, Học viện Y Dược học cổ truyền Việt Nam.
Lĩnh vực nghiên cứu: Lý thuyết đồ thị, cơ sở dữ liệu, phân tích thiết kế hệ thống an toàn thông tin, IoT, Big Data.
Email: trinhnt83@gmail.com

Vũ Chí Quang

Nhận bằng thạc sĩ Công nghệ thông tin năm 2008 tại Đại học Công nghệ - ĐHQGHN. Đang làm nghiên cứu sinh tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam.
Hiện là giảng viên chính Khoa An ninh thông tin, Học viện An ninh nhân dân.
Lĩnh vực nghiên cứu: Lý thuyết đồ thị, tối ưu hóa, cơ sở dữ liệu, IoT, Big Data, an toàn hệ thống thông tin, an ninh mạng.
Email: quangvc.hvan@gmail.com

Hà Thị Hồng Vân

Nhận bằng thạc sỹ Quản lý nhà nước về an ninh và trật tự an toàn xã hội tại Học Viện Cảnh sát năm 2013.
Hiện là Phó trưởng phòng công nghệ và giải pháp phần mềm tại Viện Công nghệ thông tin thuộc VHL Khoa học và Công nghệ Việt Nam.
Lĩnh vực nghiên cứu: Quản lý nhà nước về an ninh mạng.
Email: vanha2809@gmail.com

Nguyễn Việt Anh

Nhận bằng Tiến sĩ tại Đại học Kyoto, Nhật Bản năm 2012.
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: Machine learning, graph mining, and social net- work analysis.
Email: Email:anhnv@ioit.ac.vn