

# HỆ THỐNG GỢI Ý SỬ DỤNG THUẬT TOÁN TỐI ƯU BẦY ĐÀN

Phạm Minh Chuẩn<sup>1</sup>, Lê Thanh Hương<sup>2</sup>, Trần Đình Khang<sup>2</sup>, Nguyễn Văn Hậu<sup>1</sup>

<sup>1</sup>Khoa Công nghệ Thông tin, Đại học SPKT Hưng Yên

<sup>2</sup>Viện CNTT & TT, Đại học Bách khoa Hà Nội

chuanpm@gmail.com, huonglt@soict.hust.edu.vn, khangtd@soict.hust.edu.vn, nvhau66@gmail.com

**TÓM TẮT** - Kỹ thuật lọc cộng tác (Collaborative Filtering - CF) là một kỹ thuật gợi ý phổ biến nhất được sử dụng nhiều trong các hệ thống gợi ý đã được tích hợp trong các website thương mại điện tử (chẳng hạn như amazon.com, barnesandnoble.com, Yahoo! news, TripAdvisor.com). Kỹ thuật CF dựa trên giả thiết rằng những người dùng (user) có cùng sở thích thì sẽ quan tâm một tập item tương tự. Phương pháp phân cụm lọc cộng tác (Iterative Clustered CF - ICCF) và lập cộng tác tối ưu trọng số sử dụng thuật toán PSO (PSO-Feature Weighted) thể hiện tính hiệu quả cho hệ gợi ý mà giá trị đánh giá thuộc trong tập  $\{1, 2, \dots, 5\}$ . Tuy nhiên, các kỹ thuật đó không thể trực tiếp áp dụng cho các hệ thống gợi ý trong thực tế mà giá trị đánh giá thuộc tập  $\{0, 1\}$ . Do vậy, bài báo này đề xuất việc cải tiến hai phương pháp ICCF và PSO-Feature Weighted để có thể áp dụng được cho các hệ gợi ý mà giá trị đánh giá thuộc tập  $\{0, 1\}$ . Kết quả thực nghiệm của hai phương pháp mà chúng tôi đưa ra áp dụng trên bộ dữ liệu hệ gợi ý công việc cho thấy độ chính xác mô hình dự đoán có cải thiện rõ rệt so với phương pháp CF truyền thống đồng thời cũng giải quyết được vấn đề dữ liệu thưa mà phương pháp CF thường gặp phải.

**Từ khóa** - Hệ thống gợi ý, kỹ thuật lọc cộng tác dựa trên Item, kỹ thuật lọc cộng tác dựa trên User, phân cụm lọc cộng tác, tối ưu trọng số lọc cộng tác, thuật toán tối ưu bầy đàn, phân cụm Spectral, thuật toán k-mean.

## I. GIỚI THIỆU

Hệ thống gợi ý [1, 2] phân tích thông tin về sở thích của user về các item để cung cấp các khuyến nghị đối với các item mà phù hợp nhất với mong muốn và sở thích của người dùng. Trên thực tế, hệ thống gợi ý cố gắng thu thập những sở thích của user và mô hình hóa sự tương tác giữa user và item.

Trong kỹ thuật lọc cộng tác (Collaborative Filtering – CF), việc đưa ra những khuyến nghị về các item đối với user được xác định dựa trên những quan điểm của những user có cùng sở thích với user đó. Hệ thống lọc cộng tác biểu diễn user dựa trên những đánh giá của họ đối với tập các item. Hệ thống sẽ lựa chọn những user cùng sở thích tùy thuộc vào độ đo tương tự hoặc tương quan. Sau đó, đưa ra những dự đoán đối với những item chưa từng được user đánh giá hoặc quan tâm. Cuối cùng hệ thống sẽ gợi ý những item nào với mức độ dự đoán cao nhất cho user mục tiêu. Kỹ thuật CF đã được khẳng định sự thành công bởi rất nhiều nghiên cứu và thực nghiệm trong nhiều ứng dụng thực tế [2, 3, 4].

Nhìn chung, chất lượng của hệ thống gợi ý cộng tác có thể được tăng cường bằng cách cải thiện độ đo tương tự và việc lựa chọn tập láng giềng. Một số hạn chế chính của CF như là vấn đề dữ liệu thưa, khả năng mở rộng và thiếu dữ liệu [5, 6] có ảnh hưởng lớn đến chất lượng gợi ý. Mặc dù có nhiều nhà nghiên cứu đã cố gắng giải quyết vấn đề này, kỹ thuật lọc cộng tác vẫn cần được cải tiến nhiều hơn để cải thiện độ chính xác mô hình gợi ý. Vì kỹ thuật lọc cộng tác dựa trên những quan điểm của tập láng giềng những user có cùng sở thích với user mục tiêu, nên điều quan trọng là phải chọn tập láng giềng chính xác. Độ đo mức độ tương tự càng được cải thiện, thì việc lựa chọn láng giềng càng chính xác và gợi ý càng đúng đắn hơn.

Hiện tại, có nhiều phương pháp đã được đề xuất để cải thiện độ đo tương tự, những phương thức bao gồm độ đo PIP (Proximity-Impact-Popularity)[7], độ tương tự Union [8], Random walk counting [9], độ tương tự dựa trên phân lớp user (users–class similarity) [10], và thủ tục lặp message passing [11]; những phương pháp này đều có điểm mạnh riêng và hỗ trợ các tình huống khác nhau. Tuy nhiên, đa số các phương pháp đều tập trung trên một vấn đề cụ thể và bị ảnh hưởng bởi một vài hạn chế. Chẳng hạn, PIP đề xuất giải quyết vấn đề cold-start nhưng lại bị hạn chế bởi việc tính toán độ tương tự lọc cộng tác dựa trên user truyền thống. Độ tương tự dựa trên phân lớp user chỉ sử dụng được trong trường hợp lớp thông tin có sẵn và không nhận được kết quả ý nghĩa đối với tập dữ liệu lớn cũng như việc cập nhật độ đo tương tự được thực hiện nhiều lần. Union được sử dụng với dữ liệu thưa nhưng không có khả năng mở rộng.

Trong [12] một hệ thống gợi ý phân cụm lặp cộng tác (Iterative Clustered CF - ICCF) được đề xuất. Trong đó, phương pháp phân cụm spectral được sử dụng lặp lại trong cả hai hướng tiếp cận lọc cộng tác dựa trên user (user-based) và dựa trên item (item-based) để dự đoán những đánh giá chưa biết. Vì thế, ICCF đã thành công trong việc giải quyết vấn đề dữ liệu thưa và cold-start. Tuy nhiên tất cả user và item đều có mức độ ảnh hưởng như nhau khi tính toán độ tương tự trong khi đó độ đo tương tự cần phản ánh mức độ quan trọng của các đặc trưng khác nhau.

Một số nghiên cứu đưa ra sự cải thiện độ chính xác khi những đặc trưng được gán trọng số trong trường hợp tính toán khoảng cách [13]. Trong CF, phương pháp trọng số đặc trưng gán một trọng số đến mỗi một đặc trưng (user hoặc item) để đo mức độ quan trọng của đặc trưng như thế nào trong toàn bộ độ tương tự. Breese và cộng sự [14] phỏng theo ý tưởng của tần số văn bản ngược (inverse document frequency) để gán trọng số đặc trưng trong CF. Ý tưởng chính của tiếp cận này, gọi là tần số user ngược (inverse user frequency), đó là những item phổ biến thì không

cung cấp nhiều thông tin về sở thích thực sự của user. Vì thế trọng số của những item phổ biến được đánh giá cần phải giảm. Cùng với ý tưởng giảm trọng số đối với những item phổ biến được nhiều người biết đến được thực hiện bằng cách sử dụng phương sai trọng số [15]. Ở đây, những item có phương sai lớn hơn sẽ hỗ trợ tốt trong việc phân biệt sở thích của user, do đó, nó sẽ nhận trọng số lớn hơn. Tuy nhiên, các tác giả cũng cho rằng phương pháp này cũng giảm đáng kể trung bình lỗi tuyệt đối (Mean Squared Error-MAE) so với trường hợp không gán trọng số. Yu và cộng sự [16], giới thiệu tiếp cận lý thuyết thông tin đối với gán trọng số đặc trưng. Họ cho rằng những thông tin có tác động qua lại sẽ nhận được kết quả tốt hơn. Tác giả trong [17] cũng biểu diễn một lược đồ gán trọng số tự động khác đối với hệ thống gợi ý CF. Phương pháp này đã cố gắng tìm những trọng số gán với các item khác nhau để làm cho mỗi user gần hơn với những người láng giềng của họ và xa hơn với những người không tương đồng. Phương pháp sử dụng ý tưởng tiếp cận dựa trên mô hình (model-based approaches) và làm giảm trung bình lỗi tuyệt đối. Ngoài ra, S. H. Min và I. Han [18] đã đề xuất mô hình GA-CF giống như lược đồ trọng số đặc trưng trong kỹ thuật lọc cộng tác dựa trên user truyền thống.

Bên cạnh đó, tất cả các phương pháp gán trọng số đặc trưng, được đề xuất đến nay cố gắng nâng cao việc tính toán mức độ tương đồng mà không xem xét đến những hạn chế của CF, yếu tố ảnh hưởng nhiều đến hiệu năng của hệ thống gợi ý.

Từ quan điểm toán học, trọng số đặc trưng có thể xem như một vấn đề tối ưu không lồi phi tuyến với tối thiểu đa cục bộ địa phương (multi local) [19]. Kỹ thuật tối ưu bầy đàn (Particle Swarm Optimization - PSO) có thể tìm ra giá trị tối ưu toàn cục với thiết lập điều kiện khởi tạo đơn giản. Vì nó chỉ sử dụng các phép toán nguyên thủy do đó tiết kiệm chi phí tính toán về bộ nhớ lưu trữ và tốc độ xử lý [20].

Phương pháp PSO-Feature Weighted do Abdelwahab và cộng sự [27] đề xuất sử dụng thuật toán PSO để tìm ra một bộ trọng số tối ưu  $\omega^U$  (đại diện cho mức độ quan trọng của user) và  $\omega^I$  (đại diện cho mức độ quan trọng của item) trong việc tính toán mức độ tương đồng giữa các user và giữa các item, điều này quyết định lớn đến mức độ chính xác của mô hình dự đoán. Phương pháp này làm tăng cường quá trình phân cụm theo user và item không chỉ dựa trên thông tin phản hồi tường minh của user được biểu diễn qua ma trận đánh giá  $R_{m,n}$  (m- số lượng user, n- số lượng item), mà còn sử dụng các trọng số thể hiện cho mức độ quan trọng của mỗi user và item. Vì vậy, nó cải thiện được tập láng giềng. Ngoài ra, mô hình dự đoán được lặp lại nhiều lần để ngoại suy các giá trị đánh giá chưa biết trong ma trận  $R$ , kết quả ngoại suy bước trước được sử dụng làm dữ liệu đầu vào cho bước tiếp theo cho đến khi nhận được ma trận  $R$  tối ưu đầy hơn và từ đó giúp nâng cao độ chính xác cho mô hình gợi ý.

Hạn chế đáng kể của hai phương pháp ICCF và PSO-Feature Weighted là chúng không áp dụng được với hệ gợi ý mà ma trận đánh giá  $R$  chỉ nhận giá trị nhị phân, chẳng hạn như trong hệ thống gợi ý việc làm thì người xin việc sẽ lựa chọn những công việc để ứng tuyển, hoặc trong hệ thống gợi ý bài báo khoa học thì người dùng sẽ lựa chọn các bài báo quan tâm vào trong thư viện riêng của họ. Vì vậy, trong bài báo này chúng tôi sẽ đề xuất cải tiến cho hai phương pháp ICCF và PSO-Feature Weighted nhằm áp dụng đối với bài toán gợi ý mà ma trận đánh giá  $R$  nhận giá trị dạng nhị phân, đồng thời chúng tôi cũng điều chỉnh cách thức ước lượng giá trị  $r_{ij}$  chưa biết trong ma trận  $R$  để phù hợp với bài toán gợi ý công việc; ngoài ra, chúng tôi tiến hành xác định các trọng số khi lai ghép tuyến tính phương pháp lọc cộng tác dựa trên user và dựa trên item ( $\omega^{UF}$  và  $\omega^{IF}$ ) cùng với quá trình tìm ra bộ trọng số tối ưu đại diện cho mức độ quan trọng của các user và item trong việc tính toán độ tương đồng ( $\omega^U$  và  $\omega^I$ ) với mong muốn khai thác hiệu quả phương pháp lai ghép giữa kỹ thuật lọc cộng tác dựa trên user và dựa trên item nhằm nâng cao chất lượng của hệ thống gợi ý.

## II. GIỚI THIỆU CÁC KIẾN THỨC LIÊN QUAN

### 2.1. Phương pháp phân cụm Spectral

Phương pháp ICCF ngoại suy ra các đánh giá chưa biết trong ma trận  $R$  thông qua quá trình lặp. Trong kỹ thuật này, mô hình dự đoán sử dụng phương pháp phân cụm spectral [22] trong cả hai hướng tiếp cận lọc cộng tác dựa trên user và dựa trên item [21] và phương pháp phân cụm spectral được thực hiện theo thủ tục sau đây:

**Bước 1:** Tính độ tương đồng giữa các user và giữa các item.

$$S_{ij} = \text{Exp} \left( -\frac{\|\vec{x}_i - \vec{x}_j\|^2}{2 \times \sigma^2} \right) \quad (1)$$

Trong đó  $\vec{x}_i$  và  $\vec{x}_j$  là các véc tơ tương ứng với hàng thứ  $i$  và  $j$  trong ma trận  $R$  đại diện cho user  $i, j$  khi tính độ tương đồng giữa user  $i, j$  (và khi tính độ tương đồng giữa hai item  $i, j$  thì tương ứng với cột thứ  $i$  và  $j$  trong ma trận  $R$ ) và  $\sigma$  là tham số điều chỉnh độ lớn của tập láng giềng. Nếu  $\sigma$  nhỏ sẽ thu được một cấu hình địa phương tốt hơn đối với tập láng giềng. Tuy nhiên nếu  $\sigma$  quá nhỏ thì các điểm sẽ bị phân tách (xa nhau). Do đó, giá trị thích hợp nhất của  $\sigma$  được tính theo công thức sau [23]

$$\sigma = \sqrt{\frac{1}{n} \times \sum_{i=1}^n \min_{j \neq i} d^2(\vec{x}_i, \vec{x}_j)} \quad (2)$$

Trong đó  $d$  là khoảng giữa  $\vec{x}_i$  và  $\vec{x}_j$  và  $n$  là số các user hoặc số các item

**Bước 2:** Ma trận  $D$  (diagonal degree matrix) là ma trận đường chéo chính, trong đó các phần tử được tính toán theo công thức dưới đây:

$$d_{ii} = \sum_{j=1}^n S_{ij} \quad (3)$$

**Bước 3:** Dựa trên ma trận tương tự  $S$ , thuật toán phân cụm Spectral xây dựng một đồ thị tương đồng và tính toán ma trận Laplace  $L$  tương ứng của nó như sau:

$$L = D^{-\frac{1}{2}} \times (D - S) \times D^{-\frac{1}{2}} \quad (4)$$

**Bước 4:** Sau khi tính toán ma trận  $L$ , thuật toán phân cụm Spectral sẽ dựa trên  $k$  véc tơ riêng  $(v_1, v_2, \dots, v_k)$  ứng với  $k$  trị riêng lớn nhất đầu tiên thỏa mãn biểu thức (5)

$$L \times V = \lambda \times D \times V \quad (5)$$

**Bước 5:** Xây dựng một ma trận mới  $V \in R^{n \times k}$  với các véc tơ riêng  $(v_1, v_2, \dots, v_k)$  tương ứng với các cột của ma trận

**Bước 6:** Gọi  $y_i \in R^k$  tương ứng là các véc tơ hàng của ma trận  $V$

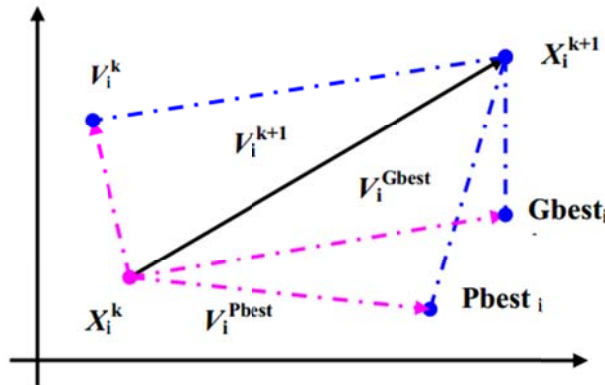
**Bước 7:** Sử dụng phương pháp phân cụm  $k$ -means để phân các điểm  $(y_i)_{i=1, 2, \dots, n}$  vào  $k$  cụm  $(C_1, C_2, \dots, C_k)$

**Bước 8:** Gán các điểm dữ liệu ban đầu  $(x_i)_{i=1, 2, \dots, n}$  theo các cụm  $C_j$  tương ứng với cụm của các điểm  $(y_i)_{i=1, 2, \dots, n}$ .

## 2.2. Phương pháp tối ưu bầy đàn (Particle Swarm Optimization - PSO)

Phương pháp tối ưu bầy đàn là một trong những thuật toán xây dựng dựa trên khái niệm trí tuệ bầy đàn để tìm kiếm lời giải cho các bài toán tối ưu hóa trên một không gian tìm kiếm nào đó. Phương pháp tối ưu bầy đàn là một dạng của các thuật toán tiến hóa quần thể, với sự tương tác giữa các cá thể trong một quần thể để khám phá một không gian tìm kiếm. PSO là kết quả của sự mô hình hóa việc đàn chim bay đi tìm kiếm thức ăn cho nên nó thường được xếp vào các loại thuật toán có sử dụng trí tuệ bầy đàn, được giới thiệu vào năm 1995 tại một hội nghị của IEEE bởi James Kennedy và Russell C. Eberhart [20]. Thuật toán này đã được áp dụng thành công trong nhiều lĩnh vực. Một ứng dụng hiệu quả về sinh học áp dụng PSO được trình bày trong [25].

PSO [26] được khởi tạo bằng một nhóm cá thể ngẫu nhiên và sau đó tìm nghiệm tối ưu bằng cách cập nhật các thể hệ. Trong mỗi thế hệ, mỗi cá thể được cập nhật theo hai vị trí tốt nhất. Giá trị thứ nhất là vị trí tốt nhất mà nó đã từng đạt được cho tới thời điểm hiện tại, gọi là  $P_{best}$ . Một nghiệm tối ưu khác mà cá thể này bám theo là nghiệm tối ưu toàn cục  $G_{best}$ , đó là vị trí tốt nhất trong tất cả quá trình tìm kiếm cả quần thể từ trước tới thời điểm hiện tại. Nói cách khác, mỗi cá thể trong quần thể cập nhật vị trí của nó theo vị trí tốt nhất của nó và của cả quần thể tính tới thời điểm hiện tại (Hình 1)



Hình 1. Sơ đồ một điểm tìm kiếm bằng phương pháp PSO

Trong đó:

$X_i^k$ : Vị trí cá thể thứ  $i$  tại thế hệ thứ  $k$

$V_i^k$ : Vận tốc cá thể thứ  $i$  tại thế hệ thứ  $k$

$X_i^{k+1}$ : Vị trí cá thể thứ  $i$  tại thế hệ thứ  $k+1$

$V_i^{k+1}$ : Vận tốc cá thể thứ  $i$  tại thế hệ thứ  $k+1$

$P_{best}$ : Vị trí tốt nhất của cá thể thứ  $i$

$G_{best}$ : Vị trí tốt nhất trong quần thể

Vận tốc và vị trí của cá thể trong quần thể được tính như sau:

$$V_i^{k+1} = \omega \times V_i^k + c_1 \times r_1 \times (P_{best} - X_i^k) + c_2 \times r_2 \times (G_{best} - X_i^k) \quad (6)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (7)$$

Trong đó:

$\omega$ : là hệ số quán tính

$c_1, c_2$ : Các hệ số gia tốc, nhận giá trị từ 1.5 đến 2.5

$r_1, r_2$ : Các số ngẫu nhiên nhận giá trị trong khoảng  $[0, 1]$

Giá trị của trọng số quán tính  $\omega$  sẽ giảm tuyến tính từ 1 đến 0 tùy thuộc vào số lần lặp xác định trước.

Các nhà nghiên cứu đã tìm ra giá trị của  $\omega$  lớn cho phép các cá thể thực hiện mở rộng phạm vi tìm kiếm, giá trị của  $\omega$  nhỏ làm tăng sự thay đổi để nhận được giá trị tối ưu địa phương. Bởi vậy, người ta đã nhận thấy rằng hiệu năng tốt nhất có thể đạt được khi sử dụng giá trị  $\omega$  lớn (chẳng hạn 0.9) ở thời điểm bắt đầu và sau đó giảm dần dần cho đến khi đưa ra được giá trị khác nhỏ của  $\omega$ .

### Thuật toán PSO

1. Khởi tạo quần thể:

(a) Thiết lập các hằng số:  $k_{max}, c_1, c_2$ .

(b) Khởi tạo ngẫu nhiên vị trí cá thể  $x_0^i$  thuộc miền  $D$  trong tập  $IR^n$  với  $i = 1, 2, \dots, s$ .

(c) Khởi tạo ngẫu nhiên vận tốc cá thể:  $0 \leq v_0^i \leq v_0^{max}$  với  $i = 1, \dots, s$ .

(d) Đặt  $k = 1$ ;

2. Tối ưu hóa:

(a) Đánh giá hàm  $f_k^i$  bằng tọa độ của  $x_k^i$  tính toán được trong không gian tìm kiếm.

(b) Nếu  $f_k^i < f_{best}^i$  thì  $f_{best}^i = f_k^i$  và  $p_k^i = x_k^i$

(c) Nếu  $f_k^i < f_{best}^g$  thì  $f_{best}^g = f_k^i$  và  $p_k^g = x_k^i$

(d) Nếu thỏa mãn tiêu chuẩn hội tụ thì dừng lại rồi thực hiện bước 3.

(e) Cập nhật tất cả các vận tốc  $v_k^i$  và vị trí  $x_k^i$

(f) Tăng  $i$ . Nếu  $i > s$  thì đặt  $i = 1$ , tăng  $k$

(g) Quay trở lại từ bước 2(a).

3. Kết thúc.

Với  $k_{max}$  là số lần lặp tối đa.

### 2.3. Phương pháp lặp cộng tác tối ưu trọng số dựa trên thuật toán PSO

Mục đích chính của phương pháp PSO-Feature Weighted là nhằm giải quyết vấn đề dữ liệu thừa và thiếu dữ liệu đối với phương pháp lọc cộng tác truyền thống. Những trọng số tương ứng với user và item được xác định bằng cách sử dụng thuật toán tối ưu bầy đàn, những trọng số này cho biết tầm quan trọng của mỗi user và mỗi item khi tính toán độ tương đồng sử dụng trong quá trình gợi ý. Những trọng số tối ưu được sử dụng để tăng cường độ đo tương đồng giữa những user và giữa những item và nó sẽ cải thiện đáng kể quá trình lựa chọn láng giềng trong bài toán phân cụm. Phương pháp này đã tích hợp việc tối ưu các trọng số trong thuật toán phân cụm lặp cộng tác để nâng cao độ chính xác của hệ thống gợi ý.

Kết quả thực nghiệm khi áp dụng phương pháp này trong hệ thống gợi ý (sử dụng dữ liệu MovieLens và Book - crossing) đã cho thấy chất lượng gợi ý đã được cải thiện đáng kể so với các phương pháp hiện tại, đồng thời cũng khắc phục một số hạn chế của những phương pháp này.

## III. CẢI TIẾN PHƯƠNG PHÁP ICCF VÀ PSO-FEATURE WEIGHTED ÁP DỤNG CHO HỆ THỐNG GỢI Ý CÔNG VIỆC

Phương pháp PSO-Feature Weighted dựa trên phương pháp lọc cộng tác, do vậy, khi áp dụng phương pháp này đối với bài toán gợi ý công việc chúng tôi chỉ quan tâm đến thông tin liên quan đến người dùng đã từng ứng tuyển công việc nào  $apply(userid, jobid)$ . Trong ma trận đánh giá  $R_{m \times n} = (r_{ij})_{m \times n}$  thì  $r_{ij} = 1$  khi người dùng  $i$  đã từng ứng tuyển công việc  $j$ , với  $m, n$  lần lượt là số lượng người dùng và công việc, như vậy mỗi người dùng được biểu diễn thông qua một véc tơ hàng của ma trận  $R$ , mỗi công việc được biểu diễn thông qua một véc tơ cột của ma trận  $R$ . Trong phần này chúng tôi trình bày sự cải tiến hai phương pháp ICCF và PSO-Feature Weighted để có thể áp dụng cho bài toán gợi ý công việc.

### 3.1. Phương pháp ICCF cải tiến (ICCF-Improved)

Phương pháp ICCF do Abdelwahab và cộng sự đề xuất đã sử dụng quá trình lặp để nội suy ra những giá trị  $r_{ij} \in \{1, 2, 3, 4, 5\}$  chưa biết trong ma trận  $R$  với mục đích giải quyết vấn đề dữ liệu thừa. Tuy nhiên, đối với bài toán gợi ý công việc thì ma trận  $R$  là ma trận dạng nhị phân, bởi vậy, không thể ước lượng trực tiếp giá trị chưa biết trong ma trận  $R$ , do đó chúng tôi đã đề xuất công thức để tính mức độ quan tâm của người dùng  $i$  đối với công việc  $j$  ký hiệu là  $p_{ij}$  sau đó dựa trên giá trị của  $p_{ij}$  chúng tôi chỉ lựa chọn ra những cặp (người dùng - công việc) mà có mức độ quan tâm lớn hơn một ngưỡng cho trước để ước lượng những giá trị  $r_{ij}$  chưa biết trong ma trận  $R$ . Cụ thể thuật toán ICCF-Improved được trình bày như sau:

- (1) Xác định cụm người dùng và công việc dựa trên phương pháp phân cụm spectral đã trình bày trong 2.1
- (2) Xác định mức độ quan tâm của người dùng đến công việc dựa trên cụm người dùng (user-based CF)

Mức độ quan tâm của người dùng  $i$  đối với công việc  $j$  ( $p_{ij}^U$ ) được ước lượng bởi công thức sau:

$$p_{ij}^U = \frac{1}{N_U} \times \sum_l Sim^U(i, l) \tag{8}$$

Trong đó,  $l$  là chỉ số của các người dùng trong cùng cụm với người dùng  $i$  mà đã ứng tuyển công việc  $j$ ,  $Sim^U(i, l)$  là độ tương đồng giữa người dùng  $i$  và người dùng  $l$ ;  $N_U$  là tổng số người dùng trong cùng cụm với người dùng  $i$  mà đã ứng tuyển công việc  $j$ .

- (3) Xác định mức độ quan tâm của người dùng đến công việc dựa trên cụm công việc (item-based CF)

Mức độ quan tâm của người dùng  $i$  đối với công việc  $j$  ( $p_{ij}^I$ ) được ước lượng bởi công thức (9)

$$p_{ij}^I = \frac{1}{N_I} \times \sum_k Sim^I(k, j) \tag{9}$$

Trong đó,  $k$  là chỉ số của các công việc trong cùng cụm với công việc  $j$  mà đã được ứng tuyển bởi người dùng  $i$ ,  $Sim^I(k, j)$  là độ tương đồng giữa công việc  $k$  và công việc  $j$ ;  $N_I$  là tổng số công việc cùng trong cụm với công việc  $j$  mà đã được ứng tuyển bởi người dùng  $i$ .

- (4) Xác định mức độ quan tâm của người dùng đến công việc

Kết hợp mức độ quan tâm  $p_{ij}^I, p_{ij}^U$  từ hai tiếp cận phân cụm Spectral dựa trên người dùng và dựa trên công việc, mức độ quan tâm của người dùng  $i$  và công việc  $j$  cuối cùng  $p_{ij}$  được tính như sau:

$$p_{ij} = p_{ij}^I + p_{ij}^U \tag{10}$$

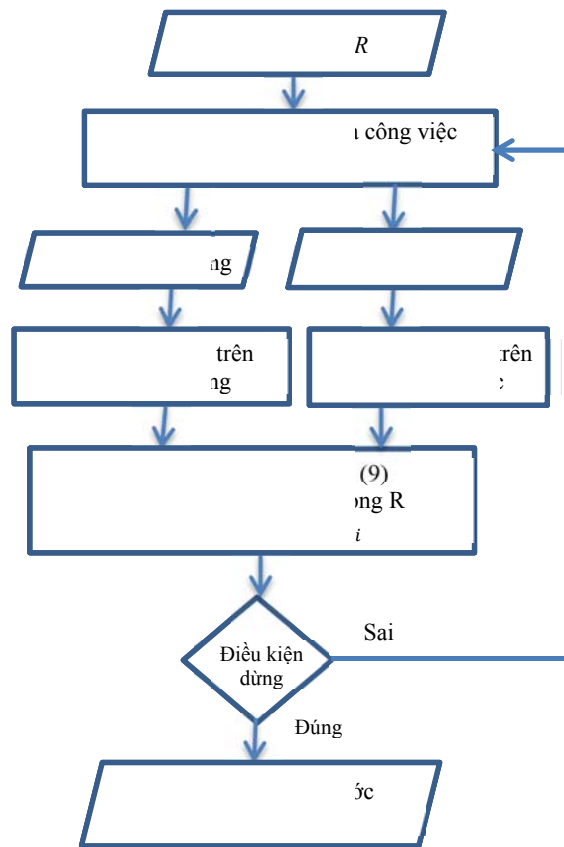
- (5) Ước lượng các giá trị chưa biết trong ma trận  $R$  dựa trên mức độ quan tâm của người dùng với công việc

$$r_{ij} = 1 \text{ nếu } p_{ij} > p_{ai} \tag{11}$$

Giá trị  $p_{ai}$  chúng tôi chọn dựa trên mức độ quan tâm trung bình của từng người dùng với tất cả các công việc mà họ đã ứng tuyển, cụ thể được tính theo công thức (12) như sau:

$$p_{ai} = \frac{1}{|R_i|} \times \sum_{k \in R_i} p_{ik} \tag{12}$$

Trong đó,  $R_i$  là tập các công việc đã được ứng tuyển bởi người dùng  $i$ . Trong phương pháp này, chúng tôi sử dụng độ cosin để tính độ tương đồng giữa hai người dùng, giữa hai công việc.



**Hình 2.** Thuật toán ICCF-Improved

**3.2. Phương pháp PSO - Feature Weighted cải tiến (PSO-FW Improved)**

Trong phần này, chúng tôi đề xuất một cải tiến cho phương pháp PSO-Feature Weighted (để cho gọn, chúng tôi sẽ gọi là PSO-FW Improved) trên cơ sở phương pháp ICCF-Improved đã trình bày ở phần trên, trong đó tập trung vào việc điều chỉnh cách thức để ước lượng các giá trị chưa biết trong ma trận  $R$  đồng thời chúng tôi cũng đề xuất việc xác định định các trọng số  $\omega^F$  và  $\omega^{UF}$  trong công thức (17) thông qua thuật toán PSO. Hình 4 cho biết thuật toán bắt đầu với việc lựa chọn các cá thể chính là các trọng số tương ứng  $(\omega, \omega^U, \omega^F, \omega^{UF})$ . Mỗi giá trị trong cá thể thể hiện mức độ quan trọng của đặc trưng tương ứng. Giá trị bằng 1 cho biết mức độ quan trọng nhất, trong khi giá trị 0 có nghĩa đặc trưng là không quan trọng và sẽ không được sử dụng trong bước tính toán mức độ tương tự (tương đồng).

**3.2.1 Cấu hình thuật toán PSO**

Các cá thể PSO được xây dựng như hình 3 được biểu diễn như  $\{\omega_1^I, \omega_2^I, \omega_3^I, \dots, \omega_n^I, \omega_1^U, \omega_2^U, \omega_3^U, \dots, \omega_m^U, \omega^{IF}, \omega^{UF}\}$  trong đó,  $\omega_j^I, j=1, 2, \dots, n$ , và  $\omega_i^U, i=1, 2, \dots, m$  tương ứng là các trọng số của công việc và người dùng;  $\omega^{IF}, \omega^{UF}$  là các trọng số tương ứng với mức độ ưu tiên của kỹ thuật gợi ý user-based CF và item-based CF sử dụng phân cụm spectral trong một cách kết hợp lại một cách tuyến tính để đưa ra kết quả gợi ý cuối cùng của mô hình và các trọng số này nhận giá trị thực trong khoảng  $[0, 1]$ . Quần thể khởi tạo là một tập ngẫu nhiên các cá thể đại diện bởi các trị bất kỳ trong không gian tìm kiếm các nghiệm ứng cử trong bài toán tối ưu. Các tham số của thuật toán được đưa ra trong bảng 1. Những tham số này điều khiển tốc độ ước lượng các trọng số tối ưu của thuật toán và cân bằng giữa việc tìm kiếm cục bộ và toàn cục.

|              |     |              |              |              |     |              |               |               |
|--------------|-----|--------------|--------------|--------------|-----|--------------|---------------|---------------|
| $\omega_1^I$ | ... | $\omega_n^I$ | $\omega_1^U$ | $\omega_2^U$ | ... | $\omega_m^U$ | $\omega^{IF}$ | $\omega^{UF}$ |
| 1            | ... | 0            | 0.5          | 0.7          | ... | 1            | 0.3           | 0.7           |

Trọng số của các công việc được sử dụng trong ma trận tương đồng người dùng
Trọng số của người dùng được sử dụng trong ma trận tương đồng công việc
Trọng số tương ứng với mức độ ưu tiên của hai kỹ thuật gợi ý

Hình 3: Ví dụ về biểu diễn cá thể cho thuật toán PSO

|   |     |
|---|-----|
| Kích thước quần thể (số cá thể)               | 15  |
| Số bước lặp tối đa (số thế hệ của quần thể)   | 100 |
| Hệ số gia tốc cục bộ (c1)                     | 2   |
| Hệ số gia tốc toàn cục (c2)                   | 2   |
| Hệ số quán tính khởi tạo                      | 0.9 |
| Hệ số quán tính kết thúc                      | 0.4 |
| Bước lặp khi hệ số quán tính đạt giá trị cuối | 80  |

Bảng 1. Cấu hình tham số của thuật toán PSO

**3.2.2 Mô hình dự đoán và hàm fitness của thuật toán PSO sử dụng trong PSO-FW Improved**

Trong phần này, chúng tôi sử dụng một mô hình dự đoán trong đó các trọng số đặc trưng  $\omega$  chính là các cá thể của thuật toán PSO. Trong mô hình này, các trọng số được sử dụng để cập nhật độ tương tự cosin như biểu thức (13) tương tự như [27] dưới đây.

$$\omega sim(\vec{x}_a, \vec{x}_b) = \frac{\sum_i \omega_i \times x_{ai} \times \omega_i \times x_{bi}}{\sqrt{\sum_i \omega_i^2 \times x_{ai}^2} \sqrt{\sum_i \omega_i^2 \times x_{bi}^2}} \tag{13}$$

Mô hình dự đoán sử dụng trong PSO-FW Improved được áp dụng theo thủ tục sau:

- (1) Phân cụm người dùng và công việc sử dụng phương pháp Spectral

Khi xây dựng ma trận tương đồng giữa các người dùng, thì cả giá trị đánh giá và trọng số của công việc được sử dụng để tính toán mức độ tương đồng giữa các người dùng. Và cũng tương tự như khi xây dựng ma trận độ tương đồng giữa các công việc, giá trị đánh giá và trọng số của người dùng cũng được sử dụng để tính toán mức độ tương đồng giữa các công việc theo công thức sau:

$$S_{ij} = Exp\left(-\frac{\|\overline{\omega x}_i - \overline{\omega x}_j\|^2}{2\sigma^2}\right) \tag{14}$$

Trong đó  $\overline{\omega x}_i = (\omega_1 \times x_{i1}, \omega_2 \times x_{i2}, \dots, \omega_m \times x_{im})$

Sau đó chúng ta chuyển đến bước tiếp theo của thuật toán phân cụm Spectral như đã đề cập ở mục 2.1, nhận được các cụm tương ứng với người dùng và công việc

(2) *Áp dụng phương pháp user-based CF dựa trên phân cụm Spectral*

Khi áp dụng thuật toán phân cụm dựa trên người dùng, mức độ quan tâm của người dùng  $i$  đối với công việc  $j$  ( $p_{ij}^U$ ) được ước lượng bởi công thức sau:

$$p_{ij}^U = \frac{1}{N_U} \times \sum_l wSim^U(i, l) \quad (15)$$

Trong đó,  $l$  là chỉ số của các người dùng trong cùng cụm với người dùng  $i$  mà đã ứng tuyển công việc  $j$ ,  $wSim^U(i, l)$  là độ tương đồng giữa người dùng  $i$  và người dùng  $l$ ;  $N_U$  là tổng số người dùng trong cùng cụm với người dùng  $i$  mà đã ứng tuyển công việc  $j$ .

(3) *Áp dụng phương pháp item-based CF dựa trên phân cụm Spectral*

Khi áp dụng thuật toán phân cụm công việc dựa trên Spectral thì mức độ quan tâm của người dùng  $i$  đối với công việc  $j$  ( $p_{ij}^I$ ) được ước lượng bởi công thức sau:

$$p_{ij}^I = N_I \times \sum_k wSim^I(k, j) \quad (16)$$

Trong đó,  $k$  là chỉ số của các công việc trong cùng cụm với công việc  $j$  mà đã được ứng tuyển bởi người dùng  $i$ ,  $wSim^I(k, j)$  là độ tương đồng giữa công việc  $k$  và công việc  $j$ ;  $N_I$  là tổng số công việc cùng trong cụm với công việc  $j$  mà đã được ứng tuyển bởi người dùng  $i$ .

(4) *Xác định mức độ quan tâm của người dùng  $i$  đối với công việc  $j$*

Kết hợp mức độ quan tâm  $p_{ij}^I$ ,  $p_{ij}^U$  từ hai tiếp cận item-based CF và user-based CF dựa trên phân cụm Spectral, mức độ quan tâm của người dùng  $i$  và công việc  $j$  cuối cùng  $p_{ij}$  được tính như sau:

$$p_{ij} = \omega^{IF} \times p_{ij}^I + \omega^{UF} \times p_{ij}^U \quad (17)$$

Trong đó,  $0 \leq \omega^{IF}, \omega^{UF} \leq 1$  được xác định qua thực nghiệm khi áp dụng thuật toán PSO.

(5) *Ước lượng các giá trị chưa biết trong ma trận  $R$  dựa trên mức độ quan tâm của người dùng với công việc theo công thức (11)*

(6) *Tính độ phù hợp (fitness) của mỗi cá thể trong thuật toán PSO*

Khi áp dụng phương pháp PSO-FW Improved cho hệ thống gợi ý công việc thì độ *fitness* của mỗi cá thể trong thuật toán PSO sẽ được tính dựa trên mức độ hội tụ của ma trận ước lượng  $R^{(k)}$  tại bước thứ  $k$ , tức là nếu ma trận  $R^{(k)}$  và ma trận  $R^{(k-1)}$  có sự khai khác càng ít thì độ *fitness* càng nhỏ; theo đó độ *fitness* của mỗi cá thể được tính theo công thức sau:

$$fitness = \frac{card(|R^{(k-1)} - R^{(k)}|)}{card(R^{(k-1)})} \quad (18)$$

Trong đó,  $card(R^{(k)})$  là số các phần tử  $r_{ij}=1$  trong ma trận  $R^{(k)}$ .

Giá trị của hàm *fitness* cho biết khoảng cách giữa vị trí hiện tại của cá thể và vị trí tối ưu. Tại mỗi bước lặp, thuật toán sẽ cố gắng giảm khoảng cách này. Bởi vậy, thuật toán trở thành quá trình cực tiểu hóa trong đó mỗi cá thể cố gắng giảm khoảng cách giữa vị trí hiện tại và vị trí tối ưu. Vì thế, nếu giá trị *fitness* bằng 0 thì vị trí hiện tại của cá thể là tối ưu.

### 3.2.3. Các bước tối ưu trọng số và dự đoán giá trị $r_{ij}$ chưa biết

Đây là phần quan trọng của thuật toán PSO-FW Improved, bao gồm các bước sau:

**Bước 1:** *Ước lượng những giá trị chưa biết  $r_{ij}$  trong ma trận  $R$  theo hai khía cạnh: dựa trên người dùng và dựa trên công việc.*

$$r_{ij} = \frac{1}{N_U} \times \sum_l wSim^U(i, l) + \frac{1}{N_I} \times \sum_k wSim^I(k, j) \quad (19)$$

Trong đó,  $N_U$  là tổng công việc do người dùng  $i$  đã ứng cử;  $N_I$  tổng số người dùng đã ứng tuyển công việc  $j$ .

**Bước 2:** *Cấu hình các tham số cho thuật toán PSO như trong mục 3.2.1*

**Bước 3:** *Với mỗi cá thể, thực hiện các công việc sau*

- Áp dụng mô hình gợi ý đã mô tả trong mục 3.2.2 để ước lượng các đánh giá chưa biết và tính độ *fitness* cho cá thể;
- Cập nhật vị trí tốt nhất của cá thể ( $Pbest$ ) theo thuật toán PSO đã mô tả trong mục 2.2

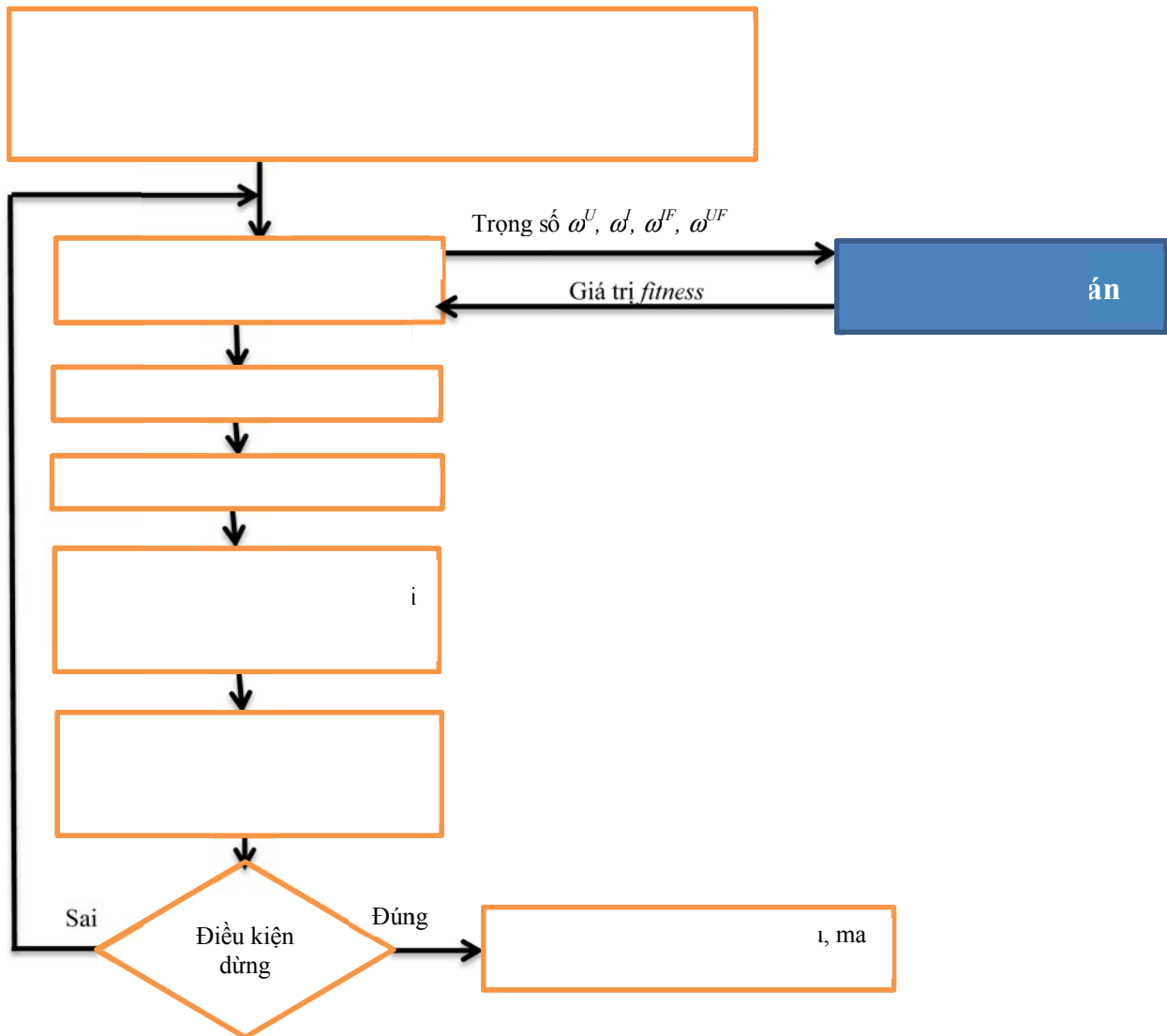
**Bước 4:** Cập nhật vị trí tốt nhất của toàn bộ quần thể (Gbest) theo thuật toán PSO đã mô tả trong mục 2.2

**Bước 5:** Cập nhật vận tốc và vị trí của mỗi cá thể theo công thức (6) và (7)

**Bước 6:** Thiết lập ma trận đánh giá  $R$  sử dụng vị trí tốt nhất của cả bầy đàn trong mô hình gợi ý và ma trận vừa thiết lập được sử dụng làm dữ liệu đầu vào cho bước tiếp theo

**Bước 7:** Nếu điều kiện dừng thỏa mãn (đạt được số bước lặp tối đa hoặc độ fitness không thay đổi) thì chuyển sang bước 8, ngược lại quay lại bước 3

**Bước 8:** Nhận được trọng số tối ưu ( $\omega^U, \omega^I, \omega^F, \omega^{UF}$ ) và ma trận đánh giá  $R$  dày hơn.



Hình 4. Lưu đồ thuật toán PSO – FW Improved

### 3.3. Thử nghiệm

#### 3.3.1 Chuẩn bị dữ liệu thử nghiệm

Chúng tôi thử nghiệm phương pháp đề xuất trên bộ dữ liệu về công việc<sup>1</sup>, bao gồm 1054 người dùng và 1682 công việc, trong đó mỗi người dùng đã từng ứng cử ít nhất một công việc, mỗi công việc được ứng cử bởi ít nhất một người dùng; dữ liệu này được biểu diễn bởi ma trận người dùng–công việc  $R$  với 1054 hàng và 1682 cột.

Dữ liệu được chia thành hai tập huấn luyện và tập kiểm tra; trong đó tập dữ liệu ban đầu được chia thành 5 phần dựa trên những cặp (người dùng, công việc ứng tuyển) trong ma trận  $R$ . Sau đó lần lượt chọn 1 phần bất kỳ làm tập kiểm tra, những phần còn lại làm tập huấn luyện.

<sup>1</sup> <http://www.kaggle.com/c/job-recommendation>



Ngoài ra, bài báo còn xem xét đến mức độ thừa của tập dữ liệu. Trong đó mức độ thừa của tập dữ liệu đối với ma trận X được tính như sau:

$$\text{Mức độ thừa } (R) = 1 - \frac{\text{Tổng số cấp (người dùng, công việc ứng tuyển)}}{\text{Tổng số cấp (người dùng, công việc)}} \quad (20)$$

Như vậy, mức độ thừa của tập dữ liệu được biểu hiện thông qua ma trận  $R = 1 - \frac{26868}{1054 \times 1682} = 0.9848$ . Với tập dữ liệu này mức độ thừa là khá cao.

### 3.3.2 Đánh giá độ chính xác mô hình gợi ý

Để đánh giá độ chính xác của mô hình gợi ý chúng ta có thể sử dụng độ đo *Precision* hoặc *Recall* hoặc là kết hợp giữa hai độ đo trên.

$$\text{Precision} = \#TP / (\#TP + \#FP) \quad (21)$$

$$\text{Recall (True Positive Rate)} = \#TP / (\#TP + \#FN) \quad (22)$$

$$F\text{-Measure} = 2x (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall}) \quad (23)$$

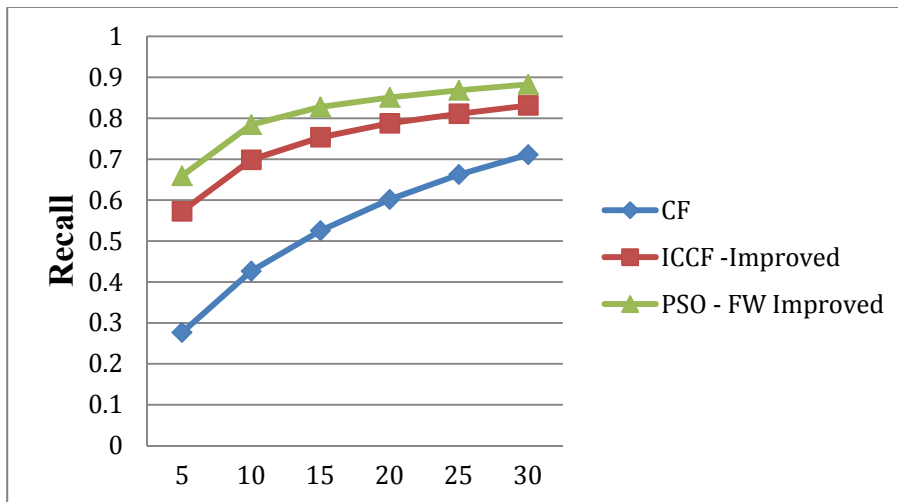
Trong đó  $\#TP$  là số công việc dự đoán đúng,  $\#FP$  số công việc dự đoán không đúng,  $\#FN$  số công việc không được dự đoán.

Trong ma trận  $R$ , giá trị  $r_{ij} = 0$  nói lên rằng hoặc người dùng không ứng tuyển công việc hoặc họ không biết về công việc đó. Điều này gây khó khăn cho việc tính toán độ chính xác (*Precision*) của mô hình dự đoán. Thay vào đó, giá trị  $r_{ij} = 1$  cho biết chắc chắn người dùng đã ứng tuyển công việc, vì vậy chúng ta sẽ sử dụng độ phủ để đánh giá mức độ chính xác của mô hình dự đoán. Theo đó, công thức về độ phủ  $\text{Recall}@N$  được tính như sau:

$$\text{Recall}@N = \frac{\text{số các công việc người dùng đã ứng tuyển trong top } N}{\text{tổng số công việc người dùng đã ứng tuyển}} \quad (24)$$

Để đánh giá toàn bộ mô hình gợi ý chúng tôi sử dụng giá trị trung bình của  $\text{Recall}@N$  tương ứng với 5 tập kiểm tra, trong mỗi một tập kiểm tra độ phủ của mô hình gợi ý là độ phủ trung bình của tất cả người dùng và giá trị này càng gần 1 thì mô hình gợi ý có độ chính xác càng cao.

### 3.3.3 Kết quả thực nghiệm



**Hình 5.** Kết quả thực nghiệm so sánh kết quả giữa các phương pháp CF, ICCF-Improved và PSO-FW Improved đối với hệ thống gợi ý công việc

Trong phần này chúng tôi sẽ tiến hành thực nghiệm dữ liệu của bài toán gợi ý công việc với các phương pháp gợi ý lọc cộng tác truyền thống, phương pháp ICCF-Improved và phương pháp PSO-FW Improved. Chúng tôi đã thực nghiệm với dữ liệu gợi ý công việc như đã mô tả trong phần trên.

Hình 5 biểu diễn độ hồi tưởng tương ứng với các phương pháp CF truyền thống, và hai phương pháp do chúng tôi đề xuất là ICCF-Improved và PSO-FW Improved lần lượt với số lượng công việc được lựa chọn để đưa ra gợi ý (TopN) là 5, 10, 15, 20, 25, và 30. Thông qua đồ thị biểu diễn trong hình 5, chúng tôi nhận thấy rằng phương pháp CF truyền thống độ chính xác thấp hơn hẳn so với hai phương pháp ICCF-Improved và PSO-FW Improved ngay cả khi giá trị TopN được tăng lên 30. Ngoài ra, đối với phương pháp ICCF-Improved mặc dù độ chính xác đã được cải thiện đáng kể so với phương pháp CF nhưng vẫn thấp hơn so với phương pháp PSO-FW Improved ứng với các giá trị TopN khác nhau. Rõ ràng sự cải tiến trong phương pháp PSO-FW Improved đã cho kết quả tốt hơn hẳn so với hai phương pháp còn lại trong toàn bộ các số lượng công việc được lựa chọn để đưa ra gợi ý. Đây là một kết quả rất đáng chú ý.

#### IV. KẾT LUẬN

Phương pháp ICCF và PSO-Feature Weighted đã giải quyết khá tốt cho hệ gợi ý mà giá trị đánh giá là các số trong tập  $\{1, 2, \dots, 5\}$ . Rất tiếc nó không áp dụng trực tiếp được cho nhiều hệ gợi ý chẳng hạn như hệ thống gợi ý bài báo, gợi ý việc làm và gợi ý tin tức; mà ở đó miền đánh giá nhận giá trị nhị phân. Để giải quyết vấn đề này, bài báo đã có hai đóng góp.

Thứ nhất, chúng tôi điều chỉnh cách thức để ước lượng giá trị đánh giá chưa biết ( $r_{ij}$ ) trong ma trận  $R$  trong cả hai phương pháp ICCF và PSO-Feature Weighted để áp dụng cho bài toán gợi ý có miền đánh giá nhị phân. Thứ hai, chúng tôi đưa ra cách tiến hành xác định trọng số khi lai ghép tuyến tính phương pháp user-based CF và item-based CF dựa trên phân cụm spectral ( $\omega^F$  và  $\omega^{UF}$ ) cùng với quá trình tìm ra bộ trọng số tối ưu đại diện cho mức độ quan trọng của các người dùng và công việc trong việc tính toán độ tương đồng ( $\omega^U$  và  $\omega^J$ ). Bằng cách như vậy, chúng tôi mong muốn khai thác hiệu quả phương pháp lai ghép giữa user-based CF và item-based CF nhằm nâng cao chất lượng của hệ gợi ý.

Chúng tôi sẽ tiếp tục nghiên cứu để cải thiện phương pháp đã đề xuất nhằm khai thác thêm thông tin mô tả của các công việc khắc phục vấn đề công việc mới tức là những công việc chưa từng được ai ứng tuyển và nâng cao chất lượng gợi ý.

#### V. TÀI LIỆU THAM KHẢO

- [1] B. N. Miller, J. A. Konstan, and J. Riedl. Toward a personal recommender system. In Proceedings of ACM Trans. Inform. Syst., 22(3), pages 437-476, 2004.
- [2] G. Adomavicius and A. Tuzhilin. Towards the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Engine, 17(6), pages 734-749, 2005.
- [3] R. Baraglia and F. Silvestri. An online recommender system for large web sites. In proceedings of IEEE/WIC/ACM International Conference on Web Intelligence, Beijing, China, pages 199-205, 2004.
- [4] J. Bobadilla, F. Serradilla, and A. Hernando. Collaborative filtering adapted to recommender systems of e-learning. Knowledge- Based Systems, 22, pages 261-265, 2009.
- [5] M. Grcar, D. Mladenic, B. Fortuna, and M. Grobelnik. Data sparsity issues in the collaborative filtering framework. In Proceedings of Advances in Web Mining and Web Usage Analysis (LNAI. 4198), pages 58-76, 2006.
- [6] M. Deshpande, G. Karypis. Item-based top- n recommendation algorithms. ACM Transactions on Information Systems, 22, pages 5-53, 2004.
- [7] H. Ahn. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences, 178(1), pages 37-51, 2008.
- [8] P. Symeonidis, A. Nanopoulos, A.N. Papadopoulos, and Y. Manolopoulos. Collaborative filtering: fallacies and insights in measuring similarity. In Proceedings of the 10th PKDD Workshop on Web Mining (WEBMine2006), Berlin, Germany, pages 56-67, 2006.
- [9] F. Fouss, A. Pirotte, J. M. Renders, and M. Saerens. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Transactions on Knowledge and Data Engineering, 19(3), pages 355-369, 2007.
- [10] C. Zeng, C. Xing, L. Zhou, and X. Zheng. Similarity measure and instance selection for collaborative filtering. International Journal of Electronic Commerce, 8(4), pages 115-129, 2004. B. Jeong, J. Lee, and H. Cho. Improving memory-based collaborative filtering via similarity updating and prediction modulation. Information Sciences, 180(5), pages 602-612, 2010.
- [12] A. Amira, H. Sekiya, I. Matsuba, Y. Horiuchi, and S. Kuroiwa. Collaborative Filtering Based on an Iterative Prediction Method to Alleviate the Sparsity Problem. In Proceedings of the 11<sup>th</sup> International Conference on Information Integration and Web-based Applications and Services (iiWAS2009), pages 373-377, Kuala Lumpur, Malaysia, 2009.
- [13] D. Wettschereck, D. W. Aha, and T. Mohri. A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. Artif. Intell. Rev., 11(1-5), pages 273-314, 1997.
- [14] J. S. Breese, D. Heckerman, and C. M. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conf. of Uncertainty in Artificial Intelligence, pages 43-52, Madison, WI, 1998.
- [15] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 230-237, Berkeley, California, United States, 1999.

- [16] K. Yu, X. Xu, M. Ester, and H. P. Kriegel. Feature weighting and instance selection for collaborative filtering: An information-theoretic approach. *Knowl. Inf. Syst.*, 5(8), pages 201-224, 2003.
- [17] R. Jin, J. Y. Chai, and L. Si. An automatic weighting scheme for collaborative filtering. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 337-344, Sheffield, United Kingdom, 2004.
- [18] S. H. Min and I. Han. *Optimizing Collaborative Filtering Recommender Systems*. Lecture Notes in Computer Science, vol.3528, pages 313-319, 2005.
- [19] D. Pudjianto, S. Ahmed, and G. Strbac. Allocation of VAr support using LP and NLP based optimal power flows. *IEE Proc. Generation, transmission and distribution*, 149(4), pages 377-383, 2002.
- [20] J. Kennedy, and R.C. Eberhart. Particle swarm optimization. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, pages 1942- 1948, 1995.
- [21] M. Papagelis, D. Plexousakis. Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents. *Engineering Applications of Artificial Intelligence*, 18(7), pages 781-789, 2005.
- [22] A. Y. Ng, M. I. Jordan, Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, vol.14, pages 849-856, 2001.
- [23] A. Afifi, T. Nakaguchi, N. Tsumura, Y. Miyake. A Model Optimization Approach to the Automatic Segmentation of Medical Images. *IEICE Trans. on Information and Systems*, E93-D(4), pages 882-889, 2010.
- [24] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. An Efficient k-Means Clustering Algorithm: Analysis and Implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.24, pages 881-892, 2002.
- [25] R. Poli. Analysis of the publications on the applications of particle swarm optimization applications. *Artificial Evolution and Applications*, 2007.
- [26] A. Lazinica, *Particle Swarm Optimization*, In-Tech, Croatia, 2009.
- [27] Amira Abdelwahab, Hiroo Sekiya, Ikuo Matsuba, Yasuo Horiuchi, Shingo Kuroiwa, "Feature Optimization Approach for Improving the Collaborative Filtering Performance Using Particle Swarm Optimization," *Journal of Computational Information Systems*, vol. 8, no. 1, pp. 435-450, 2012.
- [28] Phạm Minh Chuẩn, Lê Thanh Hương, Trần Đình Khang và Cao Xuân Bách, "Hệ thống khuyến nghị công việc", DOI 10.15625/FAIR VII.2014-0336, pages 153-159.

## RECOMMENDATION SYSTEMS USING SWARM OPTIMIZATION ALGORITHM

**Phạm Minh Chuẩn, Lê Thanh Hương, Trần Đình Khang, Nguyễn Văn Hậu**

**ABSTRACT** - The Collaborative Filtering (CF), one of the most commonly used techniques of Recommendations Systems, has been integrated in e-commerce sites (such as, amazon.com, barneandnoble.com, Yahoo! news, TripAdvisor.com). The CF approach based on the underlying assumption in which if one person, namely X, has the same opinion as another person, namely Y, on a particular issue, then X has a similar opinion with B on another issue than with another person which was chosen randomly. The Iterative Clustered CF (ICCF) and the Feature - Weighted ICCF using particle swarm optimization (PSO) methods demonstrate the effectiveness for the recommendation systems in which the assessed values are in a set of  $\{1, 2, \dots, 5\}$ . However, such techniques can't be directly applied to the recommendation systems in which the rating values belong in the set  $\{0, 1\}$ . To deal with this issue, this paper proposes some improvements to ICCF and PSO - Feature Weighted, so that these proposed methods can be applied to the recommendation systems in which the rating value are in a set of  $\{0, 1\}$ . By using these proposed for the job recommender system, experimental results show that accuracy of predictive models has considerably improved compared with the traditional CF method, and one often issue of CF, sparse problems, has been mostly solved.