SRTTU

Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

JECEI

Research paper

A New Clustering Algorithm for Attributive Graphs through Information Diffusion Approaches

***S. Kianian*[1], *S. Farzi*[2, *], *H. Samak*[2]**

[1]Faculty of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran.

[2]Faculty of Computer Engineering, K. N. Toosi University of Technology, Tehran, Iran.

Article Info

Article History:
Received 02 September 2019
Reviewed 07 November 2019
Revised 13 December 2019
Accepted 13 March 2020

Keywords:
Attributive graph
Clustering algorithm
Signal similarity
Heat diffusion
Community detection

[*]Corresponding Author's Email Address:
Saeedfarzi@kntu.ac.ir

Abstract

Background and Objectives: Simplicity and flexibility constitute the two basic features for graph models which has made them functional models for real life problems. The attributive graphs are too popular among researchers because of their efficiency and functionality. An attributive graph is a graph the nodes and edges of which can be attributive. Nodes and edges as structural dimension and their attributes as contextual dimension made graphs more flexible in modeling real problems.
Methods: In this study, a new clustering algorithm is proposed based on K-Medoid which focuses on graph's structure dimension, through heat diffusion algorithm and contextual dimension through weighted Jaccard coefficient in a simultaneous matter. The calculated clusters through the proposed algorithm are of denser and nodes with more similar attributes.
Results: DBLP and PBLOG real data sets are applied to evaluate and compare this algorithm with new and well-known cluster algorithms.
Conclusion: Results indicate the outperformers of this algorithm in relation to its counterparts as to structure quality, cluster contextual and time complexity criteria.

Introduction

The simple graphs are consisting of nodes and edges collectives which indicate the graph's structural dimension. The simple graphs are applied widely in modeling things and with different counter dependencies, like as friendship and kinship in verity of efficiency realms like analyzing social network, wide web networks and sensor networks. Though, attributive graphs are simple graphs where nodes and edges can have particular features; these features indicate the graph's contextual dimension. Attributive graphs are applied as basic models in running assessments in human interactions in social systems. The graph's structural feature is indicative of individuals and communities in social science while the contextual features is indicative the individuals features' and communities thereof determining social distinction is contributive in constructing a functional graphs evaluation [1]. Today, due to public functional applications, like indicating important modules in biological graph [2] [3][4], data collection related to web pages [5] and determining events/orientations in social networks [6] [7] are of major concern. Traditional algorithms are using structural features to identify communities. The structural features, exhibited through attributive nodes and edges, are beneficial for compact connected components' distinction, like communities. However, in real world networks, node's/edge's features

Doi: 10.22061/JECEI.2020.7190.366

are too important in studying contextual function's and network's evolution. The attributive graph is an extended graph where in studying in attributive nodes and edges are applied. The node's features are indicative of particular attributions for node's contextual and semantic descriptions. In a similar sense, an edge's features introduce the type of the connection among them. An attributive graph presents a partial model reached in real world network instead of a traditional one [8]. In today's practice, the traditional clustering graph methods are being applied in attributive graphs clustering. The focus of methods is on either topological structural or on graph contextual features where each one of the clusters containing homogenous nodes with respect to the nodes and edges features, while, many recent methods apply a combination of structural and contextual information in attributive graphs clustering [9][10][11].

Considering the definition of attributive graph, the attributive graphs' clustering algorithm, as a compact connected nodes classification, is defined through homogenous features volume. The most important challenge here is to find the harmony between structural and contextual similarities of clusters' nodes and edges [12]. In this study, a new clustering algorithm is proposed for weighted attributive graphs. This algorithm is based on K-Medoid clustering algorithm where both the structural and contextual graph information are applied. A balancing factor is applied in order to balance the structural and contextual data on clustering results. The main idea is transmission of weighted attributive graphs into spatial space where the structural and contextual similarities among nodes are unified. In this proposed algorithm, a similarity criterion, based of heat diffusion [13] is employed with the objective of structural information integration, and, Jaccard weighted similarity criterion [14] is applied for contextual information integration, thus the points in new spatial space would include unified structural and contextual information. Now, K-Medoid algorithm proposed for clustering in spatial space can be applied in this space. The level of structural and contextual features' contribution can be regulated through the balancing factor [16]. Contrary to [11], clusters count and initial main seeds constitute the two basic parameters which should be determined before clustering algorithm is applied. The available algorithms apply nodes degree to find initial seeds in K-Medoid.

Here, clusters count is determined by user and is known as one of the parameters for this proposed algorithm, while the initial seeds are determined based of their degree centrality. Here, the initial seeds selection takes shorter time. Eventually, the proposed algorithm optimizes an objective function which maximizes the inner cluster similarities and minimizes the intra cluster similarities.

The two real data sets, PBLOG [10] including 1490 nodes and DBLP [10] including 10000 nodes, are utilized for experimental evaluation. Many evaluation scenarios are followed to analyze this proposed algorithm. In addition to assessing the algorithm parameters, this algorithm is compared with five other advanced: S-cluster [9], W-cluster [9], SA-cluster [9], SI-Cluster, and KSNAP [16] algorithms based on density and entropy criteria. The results here indicate the outperformers of this algorithm on its counterparts, although in some cases the obtained results are comparable. As to running time complexity, this algorithm outperforms its counter parts as well.

The rest of this paper structure is as follows: literature review is presented in Sec.2; the issues are explained Sec.3; the method is described in Sec.4; empirical studies are presented in Sec.5 and the article concluded in Sec.6.

Literature Review

Most of the graphs clustering algorithms focus on structural aspects based of different objective functions [17][30], as normalized cuts [18] and overall density [19[. The outputs of these algorithms are clusters of high density, but in these algorithms the node features of are ignored.

Today, the graph clustering algorithms focus mostly on structural and contextual aspects of an attributive graph [28][29]. Metis and Markov clustering applied CODICIL [20] clustering for content combination and similar links. The possibility of an edge belonging to one cluster is applied in estimating link similarity, while Jaccard coefficient is applied in estimating contextual similarity.

SI-Cluster [11] is a clustering algorithm based on signal similarity which introduces weighted Jaccard similarity for combining structural and contextual similarities.

Similar to SI-Cluster, this algorithm applies a balancing factor to establish equilibrium among the structural and contextual attributes. Contrary to SI-Cluster method, this applies signal similarity transmission to provide structural information.

Heat diffusion is utilized in this algorithm. Here a simpler and faster method is applied in order to find the initial main seeds, while in SI-Cluster algorithm a complex and time consuming method is employed. SA-cluster [9] is a random walk based clustering method which introduces distances unification for structural and contextual integration. The given graph is clustered based on specific count of clusters. In comparison with SA-cluster, S-cluster is introduced by [19], is of higher structural similarities and lower contextual similarities. The focus of KNSAP algorithm [16] is on contextual aspect which accumulates nodes of similar features in one cluster.

The Issue

An attributive weighted directed graph G(V,E,C,A,W), where $V = \{v_1.v_2.\dots.v_{|v|}\}$ is the set of nodes, $E = \{\langle v_i.v_j\rangle | v_i.v_j \epsilon V\}$ is the set of directed edges with weighted function of $C_{ij} \epsilon C$ and $P = \{p_1.p_2.\dots.p_{|p|}\}$ is a $|p|$ number of features for nodes' description. p_q feature is related to a v_i node, $p_q(v_i)\ \epsilon P$, with domain of $d_q = |Dom(p_q)|$ vector attached to the vi through $Wq(v_i) \in W$ weight. Thus node v_i has a feature vector $\vec{P}(v_i)$ with $|\vec{P}(v_i)| = \sum_{i=1}^{m} d_i$.

A directed weighted collaborative graph where the nodes introduce the writers' and the edges introduce the colleagues of an article is drawn in Fig. 1. For every writer, 2 adjectives of research interest and mother language are of concern.

Clustering of a directed weighted attributive graph is partitioning the given graph into k subgraph $G^k = (V^k, E^k, C^k, A^k, W^k)$ where $V = \cup_{k=1}^{K} V^k$, $E = \cup_{k=1}^{K} E^k$, $V^i \cap V^j = \emptyset$ for any $i \neq j$.

Finding an appropriate equilibrium for optimizing two independent objectives or even a contradictory one is subject to: 1) Structural objective: the inner cluster nodes are similar structurally and different among clusters and 2) Contextual objective: the inner nodes are similar contextually and different among clusters.

In order to balance these two objectives, a balancing factor, $\lambda \epsilon [0.1]$, combines both functions linearly.

$$O_f = \lambda \times O_{str} + (1 - \lambda) \times O_{con} \qquad (1)$$

where, Ostr and Ocon are structural and contextual objective, respectively.

Two important questions must be answered in order to design a clustering algorithm:

1) How are the structural and contextual similarities calculated?

2) How is the objective function optimized?

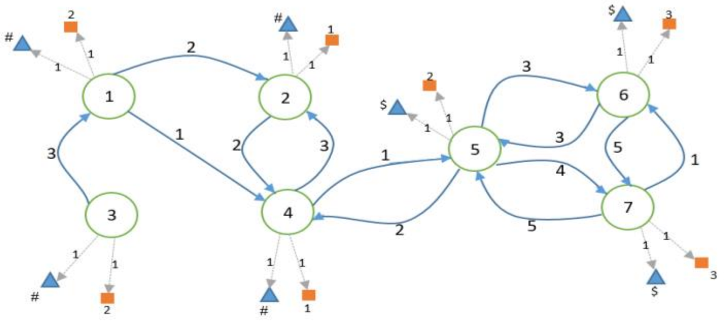


Fig. 1: An example for a directed weighted collaborative graph.

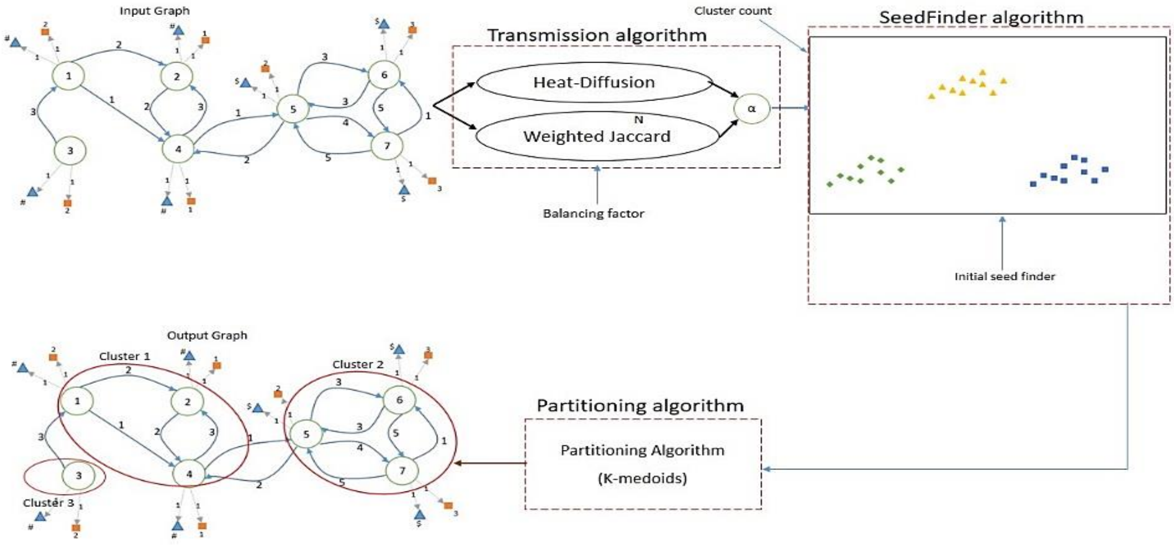


Fig. 2: System's architecture.

To answer the first question, the structural similarities are calculated through heat diffusion and contextual similarities are calculated through weighted Jaccard coefficient. Balancing factor is applied in order to integrate the obtained information and provide a new space. Points in this new space include graph's structural and contextual information. As to answer the second question, due to simplicity and low implementation time algorithm, K-Medoid algorithm is applied in this algorithm to optimize Objective function. The clusters and initial seeds count are the two main challenges which must be of concern. Cluster count is determined by the user, while for key initial seeds selection, the nodes with higher degree would be selected as initial seeds.

The Proposed method

As shown in Fig. 2, the proposed algorithm, here after H-cluster is considered a directed weighted attributive graph as an input and a partitioned graph as output. H-cluster consists of three main parts: 1) Transition algorithm 2) the initial seeds finder algorithm and 3) partitioning algorithm. Transition algorithm, converts the structural and contextual information into spatial space. To do so a heat diffusion similarity [13] and a weighted Jaccard similarity [14] are applied in measure the structural and contextual information. The initial seeds are the main challenge for clustering algorithm [21]. To come up with this problem, the seed-finder algorithm is applied in the second part of the algorithm. Partitioning algorithm finds the optimized clusters through initial main seeds finding and determines their count. The objective function is a defined through the combined structural and contextual similarities. Transition algorithm, seed-finder and partitioning algorithm will be discussed further.

A. Transition Algorithm

In this algorithm, first, the graph's structural data is calculated through heat diffusion simulation, and the contextual information graph is calculated through weighted Jaccard coefficient. At the end a linear combination of these two criteria is calculated through balancing factor.

- Structural transition

Structural transition is a graph model based on heat diffusion [13]. This model can be implemented on both directed and un-directed graphs. Heat diffusion is a physical phenomenon. In an environment, heat always flows from a point with higher temperature to a point with lower temperature. Recently, heat penetration based methods are applied in different aspects like as classification and dimensions reduction [22] [23] [24]. In this article, heat diffusion is applied for modeling the structural similarity in attributive graphs.

To transfer structural and topological data presented through a given attributive graph to spatial space, heat diffusion [13], which is a popular algorithm for data transition, is applied with the objective of calculating similarities between two nodes. According to [25], heat diffusion similarity are more efficient compared to other likewise similarity functions such as Jaccard and Cosine. Heat diffusion is main base for heat diffusion similarity calculation. To calculate the heat diffusion similarity for a graph with n number of nodes, every node is considered as a heat source. During an iterative process, each node is selected as an initial heat source to stimulate all other nodes in the given graph. The process begins with attributing one heat unit to node. After heat diffusion, the initial node and all its neighbors, store the heat in a vector of n number of dimension and re-transfer it to all neighbors. After one step (f(1) time), the source's nodes effect on the whole chart with and the received heat volume in the n dimension vector of nodes is calculated. The mentioned heat diffusion method can be described as a simple and clear mathematical process according to (2)- (9)

$$\frac{f(t+\Delta t)}{\Delta t} = \alpha(H-D)f(t) \tag{2}$$

where

$$H_{ij} = \begin{cases} 1\ (v_i\,, v_j)\epsilon\ E \text{ or } (v_j, v_i)\epsilon\ E \\ \quad 0\ i=j \\ \quad 0 \text{ otherwise} \end{cases} \tag{3}$$

and

$$D_{ij} = \begin{cases} d(v_i) \quad i=j \\ \\ 0 \text{ otherwise} \end{cases} \tag{4}$$

where, d(vi) is the degree of node vi. The D matrix is a diagonal matrix.

Thus, better exhibition all D and H matrix inputs are normalized based of each nodes degree. D and H matrices can be normalized through the following equations:

$$H_{ij} = \begin{cases} \frac{1}{d(v_i)}, (v_i, v_j)\epsilon\ E \\ \quad 0\ i=j \\ \quad 0 \text{ otherwise} \end{cases} \tag{5}$$

and

$$D_{ij} = \begin{cases} \quad 1\ i=j \\ \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

The following differential equation is applied to solve this problem:

$$\frac{d}{dt}f(t) = \alpha t(H-D)f(t) \tag{7}$$

To solve this issue we have:

$$\hat{S} = e^{a(H-D)}f(1) \tag{8}$$

where, d(v) is the node's v degree, and $e^{a(H-D)}$must be yielded through Eq.9:

$$e^{a(H-D)} = I + \alpha(H-D) + \frac{\alpha^2}{2!}(H-D)^2 + \frac{\alpha^3}{3!}(H-D)^3 + \cdots \qquad (9)$$

Eventually, the normalized matrix $\hat{S}$ includes heat diffusion data transmitted among different nodes.

The $e^{a(H-D)}$ matrix is named the diffusion core, which is repetitive heat diffusion after initial diffusion.

- Contextual transition

The Jaccard similarity is a common function applied widely on calculating the similarities among two sets. Due to the weighted attributives of the considered graph, the weighted Jaccard similarity function [26] is applied. Thus, in order to convert the contextual information, through attributive graphs, into the spatial space of n dimensions, the weighted Jaccard weighted similarity [26] is calculated between two vectors of attributive nods through (10):

$$\hat{C}_{ij} = \frac{\sum_{q=1}^{\sum_{i=1}^{m} d_i} \min\left(\vec{P}_q(i), \vec{P}_q(j)\right)}{\sum_{q=1}^{\sum_{i=1}^{m} d_i} \max\left(\vec{P}_q(i), \vec{P}_q(j)\right)} \qquad (10)$$

where $\vec{P}_q(i)$ and $\vec{P}_q(j)$ are the non-negative features vectors of nodes i and j, respectively and Max(…) and min(…) are the maximum and minimum functions, respectively.

- The proposed transition algorithm

The transition algorithm is adopted to integrate a spatial space through contextual and structural data combination. To make this spatial space, a combination of normalized linear structural and contextual similarity is applied as follows:

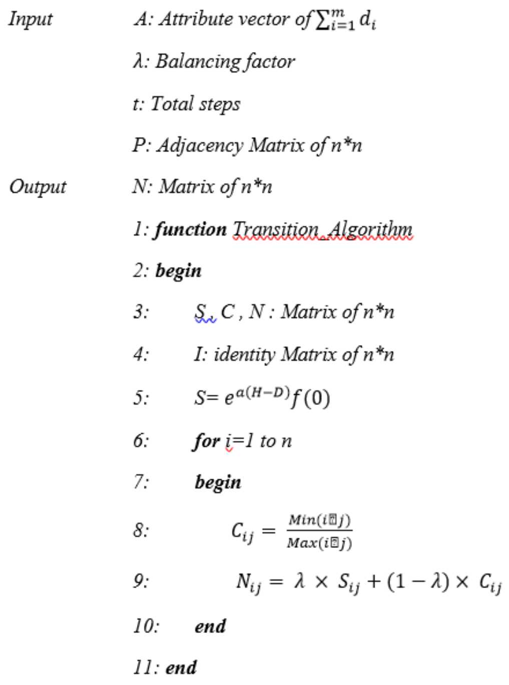
$$N_{ij} = \lambda \times \hat{S}_{ij} + (1-\lambda) \times \hat{C}_{ij} \qquad (11)$$

here, Nij calculates the contextual and structural similarities between i and j nodes. The transition algorithm is exhibited in Fig. 3.

Max(…) and Min(…) are the maximum and minimum functions, respectively, described in (10).

B. Parameter Determination

High degree nodes in a cluster play most important roles. As to they usually are considered as headers these nodes are surrounded by nodes with lower degree [11]. Thus, in order to find the cluster's initial seeds, first, the nodes must be sorted in descending degree and then, αk numbers of nodes with higher degree are selected, where k is the number of clusters and α is per-defined constant value. The first node is selected as the initial seed. The next node will be selected when it keeps the greatest distance from previously selected nodes as seeds. This process continues until k initial seeds is determined, the time complexity of this algorithm is $O(|V|\log|V| + |V|k) \cong O(|V|\log|V|)$.



```
Input     A: Attribute vector of $\sum_{i=1}^{m} d_i$
          λ: Balancing factor
          t: Total steps
          P: Adjacency Matrix of n*n
Output    N: Matrix of n*n
function Transition_Algorithm
begin
    S, C , N : Matrix of n*n
    I: identity Matrix of n*n
    S= $e^{a(H-D)} f(0)$
    for i=1 to n
    begin
        $C_{ij} = \frac{Min(i\square j)}{Max(i\square j)}$
        $N_{ij} = \lambda \times S_{ij} + (1-\lambda) \times C_{ij}$
    end
end
```

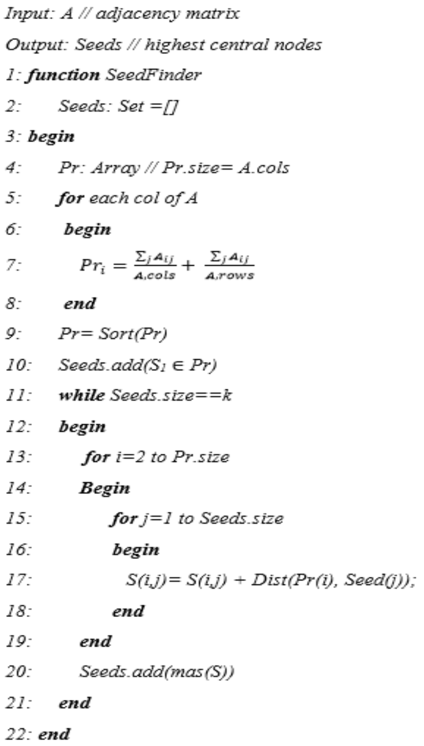
Fig. 3: Transition algorithm.

Though, the nodes with higher degree are more important and most probably are being as seed, selecting two nodes with higher degree next to each other is not a good idea. Here, the node with higher degree is selected as a seed and the nodes with lower degree are ignored. Seed-finder algorithm implementation is as follows:

Dist(…) represents the distance between the candidate node (Pri) and node j which is selected as a seed previously and S(…) is the matrix of total distances between candidate nodes and all that were selected and confirmed seeds.

C. Partitioning algorithm

The objective is to bring the inner cluster similarity to maximum and minimize the inter-cluster similarity. In order to propose a compatible partitioning algorithm, two issues must be of concern: 1) the manner of contributing each of the nodes to a cluster and 2) the manner of defining an intra-cluster objective function for having access to intra-clusters higher structural and contextual similarities and lower enter-cluster structural and contextual similarities. As to issue one, a K-Medoid based compatible partitioning algorithm is introduced [27], which is an extended version of K-Means clustering algorithm where the central seeds in a cluster are selected by Medoids. In K-Medoid based partitioning algorithm, Medoids are selected through the initial key seeds, which are identified by seed finder algorithm. During each iteration, the seed are determined once more and the points are reallocating to the clusters. The process continues while the seeds stop changing in next iteration.



```
Input: A // adjacency matrix
Output: Seeds // highest central nodes
function SeedFinder
      Seeds: Set =[]
begin
      Pr: Array // Pr.size= A.cols
      for each col of A
       begin
          $Pr_i = \frac{\sum_j A_{ij}}{A.cols} + \frac{\sum_j A_{ij}}{A.rows}$
       end
      Pr= Sort(Pr)
      Seeds.add(S1 ∈ Pr)
      while Seeds.size==k
      begin
         for i=2 to Pr.size
         Begin
             for j=1 to Seeds.size
             begin
               S(i,j)= S(i,j) + Dist(Pr(i), Seed(j));
             end
         end
         Seeds.add(mas(S))
      end
end
```

Fig. 4: The initial seed finder algorithm.

As to second issue, the density and entropy volume are applied, respectively, to calculate the structural and contextual similarities. Thus, the objective function is calculated through (2):

$$O_f = \lambda \times O_{str} + (1 - \lambda) \times O_{con} = \lambda\, D\left([G^j]_{j=1}^{j=k}\right) + (1 - \lambda) * 1/E([G^j]_{j=1}^{j=k}) \qquad (12)$$

where, λ represents the balance factor D(...) is a density function applied in structural similarity calculation and the contextual similarity is calculated through E(...), the entropy function.

The density function reveals the intensity of coherency of clusters' compression. The partitioning graph's overall density is yielded through (13):

$$D\left([G^j(V^j, E^j)]_{j=1}^{j=k}\right) = \sum_{j=1}^{k} \frac{\sum_{\langle v_p, v_q\rangle \in E^j} C(\langle v_p, v_q\rangle)}{\sum_{\langle v_p, v_q\rangle \in E} C(\langle v_p, v_q\rangle)} = \frac{1}{\sum_{\langle v_p, v_q\rangle \in E} C(\langle v_p, v_q\rangle)} \times \sum_{j=1}^{k} \sum_{\langle v_p, v_q\rangle \in E^j} C(\langle v_p, v_q\rangle) \qquad (13)$$

where, C(...) is the graph's cost function.

Also the total cluster entropy is applied to compute relevancy of nodes of a given graph with respect to attribute values, Eq. 14.

$$E\left([G^j(V^j, E^j, A^j)]_{j=1}^{j=k}\right) = \frac{1}{k}\sum_{j=1}^{k} E^c(G^j(V^j, E^j, A^j)) \qquad (14)$$

$$E^c(G^j(V^j, E^j, A^j)) = -\frac{1}{m}\sum_{q=1}^{m} \sum_{a_q \epsilon\, d_q} P(a_q, V^j) \log P(a_q, V^j) \qquad (15)$$

$$P(a_q, V^j) = \frac{|v_h \in v^j|\, a_q(h) = a_q|}{|v^j|} \qquad (16)$$

The more the placement of nodes with similar features in a cluster, the lowers the entropy.

Empirical studies

Functionality of this method is evaluated through many experiments. The three main scenarios consist of: 1) assessing the convergence of the proposed algorithm, 2) assessing the parameters' effect on cluster's quality, and running time complexity and 3) analyzing the density and entropy of this method compared to other well-known clustering algorithms.

A. Data

Two real data sets called (PBIOG) [10] and (DBLP) [10] are applied in order to evaluating the efficiency of introduced method, Table 1.

Table1. Several statistics related to data collections

	PBLOG	DBLP
Description	Political Weblog network	Co-authorship network
Nodes	1490	10000
Edges	33433	65734
Attibutes (values)	Political leaning	Prolific Primary topic

B. Configuration

This experiment is run on Intel® coreTM with seven 2.80 GHz cores and 6 G main RAM. The code is implemented MATLAB. This H-cluster method is compared with five contemporary algorithms of:, SA-Cluster, S-Cluster, W-Cluster, SI-Cluster[1] and KNSAP.The SA-Cluster algorithm is introduced by [9], where both the structural and contextual aspects of the network are of concern. The nodes closeness is calculated through random walk S-Cluster algorithm [9] where only the structural aspect is of concern. W-Cluster is presented by [9]. where the both the structural and contextual aspects are combined through a linear combination [9]. KNSAP is introduced by [16] and SI-Cluster is proposed by [11], where though both, the aspects, are combined

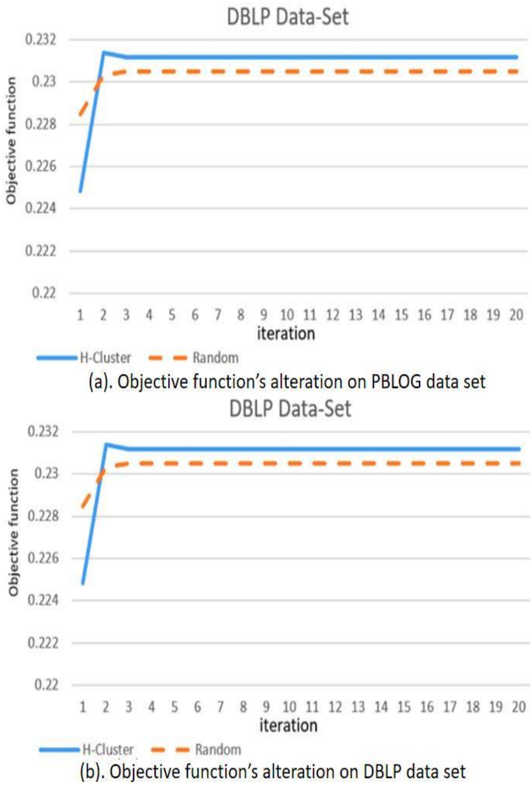
[1] SI-Cluster using signal diffusion without Seed Finder algorithm like [11]

through a linear combination accumulated the nodes with similar features in a cluster. To compare the function and efficiency of these algorithms the density and entropy criterion are calculated through (14) and (15).

C. Convergence

The objective function of every iteration of H-Cluster with or witout seed finder algorithm on PBLOG and DBLP data collection is shown in Fig. 5. The H-Cluster without seedfinder applies random function to find the seeds. The quantities are expressed as follows: (Ma teration=20 , λ = 0.5,$\alpha = 40$ (k=10 for DBLP and k=9 for PBLOG))

As observed in Fig. 5-a H-Cluster reaches convergence faster than other. As observed in Fig. 5, the maximum objective volume for both DBLP and PBLOG data sets are obtained, at the third iteration that is the initial main seeds determined based on seed finder algorithm provides the means for clustering algorithm to obtain better results in little iteration.



(a). Objective function's alteration on PBLOG data set

(b). Objective function's alteration on DBLP data set

Fig. 5: Objective function alterations at every iteration.

The two main parameters are analyzed as follows: α $(alfa)$ which applied in finding initial seeds and λ which is a structural and contextual balancing factor. The λ effect on cluster quality on DBLP data set when $\alpha = 40$ and the number of clusters is 10. Fig. 6.

Fig. 7 and Fig. 8 show Alfa's (α) effect on cluster's quality on (density, entropy and objective function) in PBLOG and DBLP datasets respectively. Here $\lambda = 0.5\ \mathrm{K(PBLOG)} = 3$ and $\mathrm{K(DBLP)} = 10$.

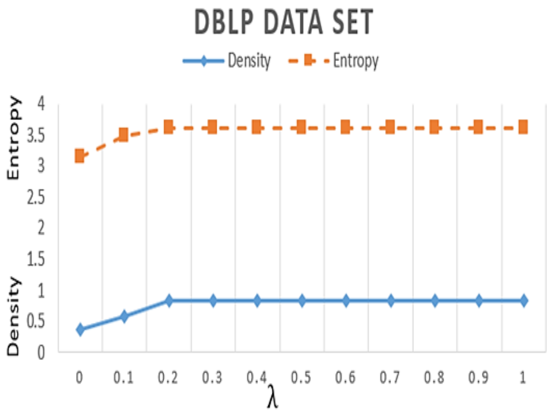


Fig. 6: Balancing factor effect on density and entropy criterion in DBLP data set.

As observed in Fig. 7(a) and Fig. 8(a) the higher the α volume the more the density. The best volume of entropy on PBLOG dataset, $(\alpha) = 5$, Fig. 7(b) and according to in Fig. 8 the best volume of entropy for DBLP dataset is $(\alpha) = 40$. After balancing the density and entropy criterion, as it Fig. 7(c) and Fig. 8, the volume of objective function is obtained. The maximum volume of objective function in PBLOG dataset is $(\alpha) = 5$ and the same volume for DBLP is $(\alpha) = 40$.

D. Running Time

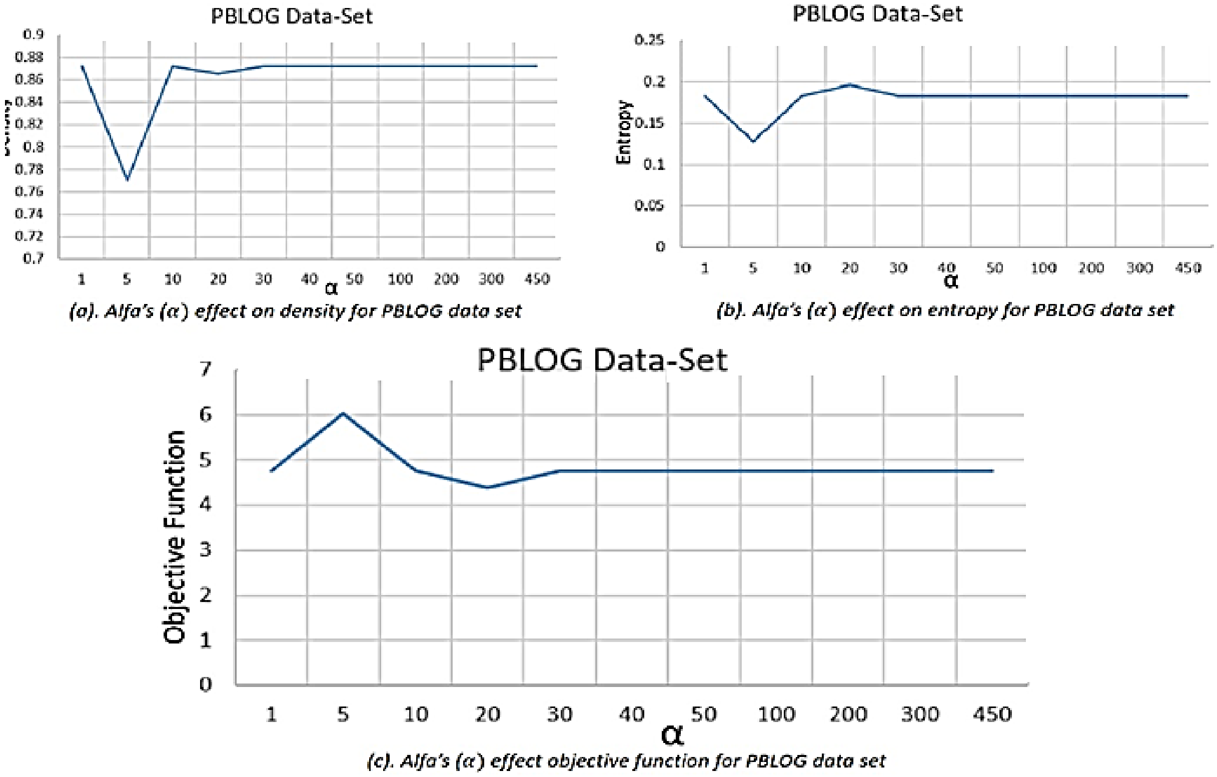
Here, Alfa (α) is the most effective factor in time complexity. The H-Cluster running time with respect to Alfa's (α) alternations in DBLP data sets illustrated in Fig. 9.

As observed in Fig. 9, an increase α volume, increases the algorithm's running time, leading to an important in cluster's quality Fig. 10.

E. Cluster Algorithms Comparisons

To compare H-cluster algorithm with other new clustering algorithms, several evaluations are run on PBLOG and DBLP datasets. Here, λ=0.5 for S-Cluster, SA-Cluster, SI-Cluster and H-Cluster and α=5k for H- cluster, SI-Cluster on PBLOG dataset with α=40k for these algorithms for DBLP dataset. The laboratory experiments' results on density and entropy in PBLOG dataset are expressed in Fig. 10.

In average, SI-Cluster (-0.01042 | +0.0887 | +3.6060) and (+0.21356 | +0.08412 | +14.549) outperforms SA-Cluster and H-Cluster as to (density | entropy | objective function) criteria, respectively. In average, H-Cluster, as to the (density | entropy | objective function) (-0.2240 | +0.00458 | -10.9431) (-0.224 | -0.08412 | -14.4590) fails compered to SA-Cluster and SI-Cluster, respectively. As observed in Fig. 10, considering the fact that both structural and contextual aspects are of concern in H-Cluster, indicating that H-Cluster as to PBLOG dataset is not as efficient as SI-Cluster, the SI-Cluster outperforms H-cluster on PBLOG dataset.

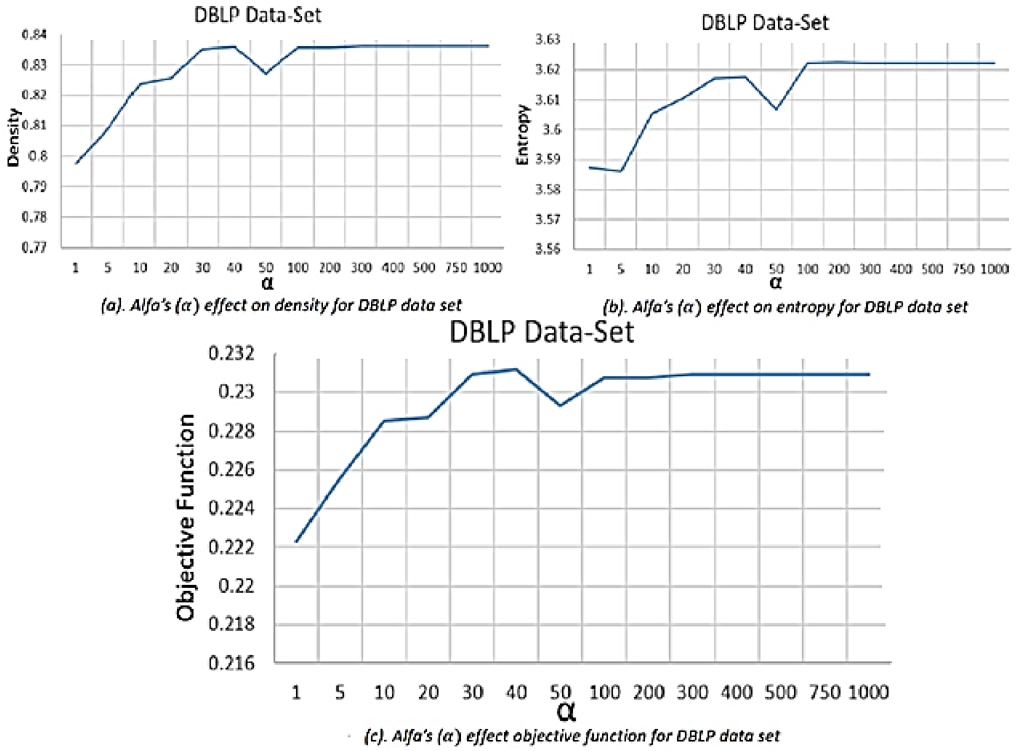


(a). Alfa's (α) effect on density for PBLOG data set

(b). Alfa's (α) effect on entropy for PBLOG data set

(c). Alfa's (α) effect objective function for PBLOG data set

Fig. 7: Alfa's (α) effect on density and entropy criterion and objective function of (PBLOG).



(a). Alfa's (α) effect on density for DBLP data set

(b). Alfa's (α) effect on entropy for DBLP data set

(c). Alfa's (α) effect objective function for DBLP data set

Fig. 8: Alfa's (α) effect on density and entropy criterion and objective function of (DBLP).

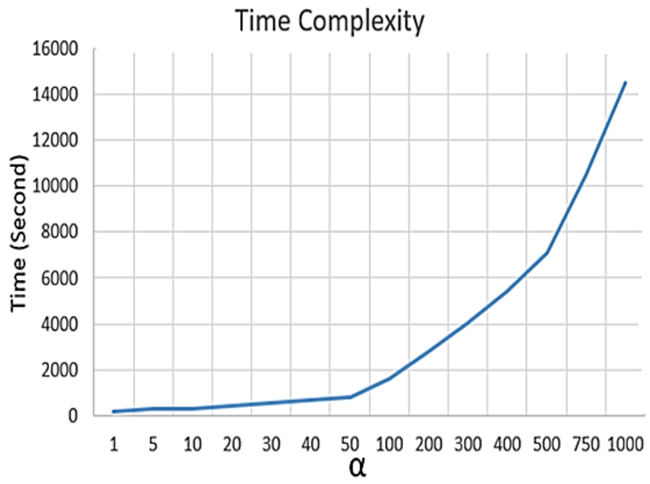
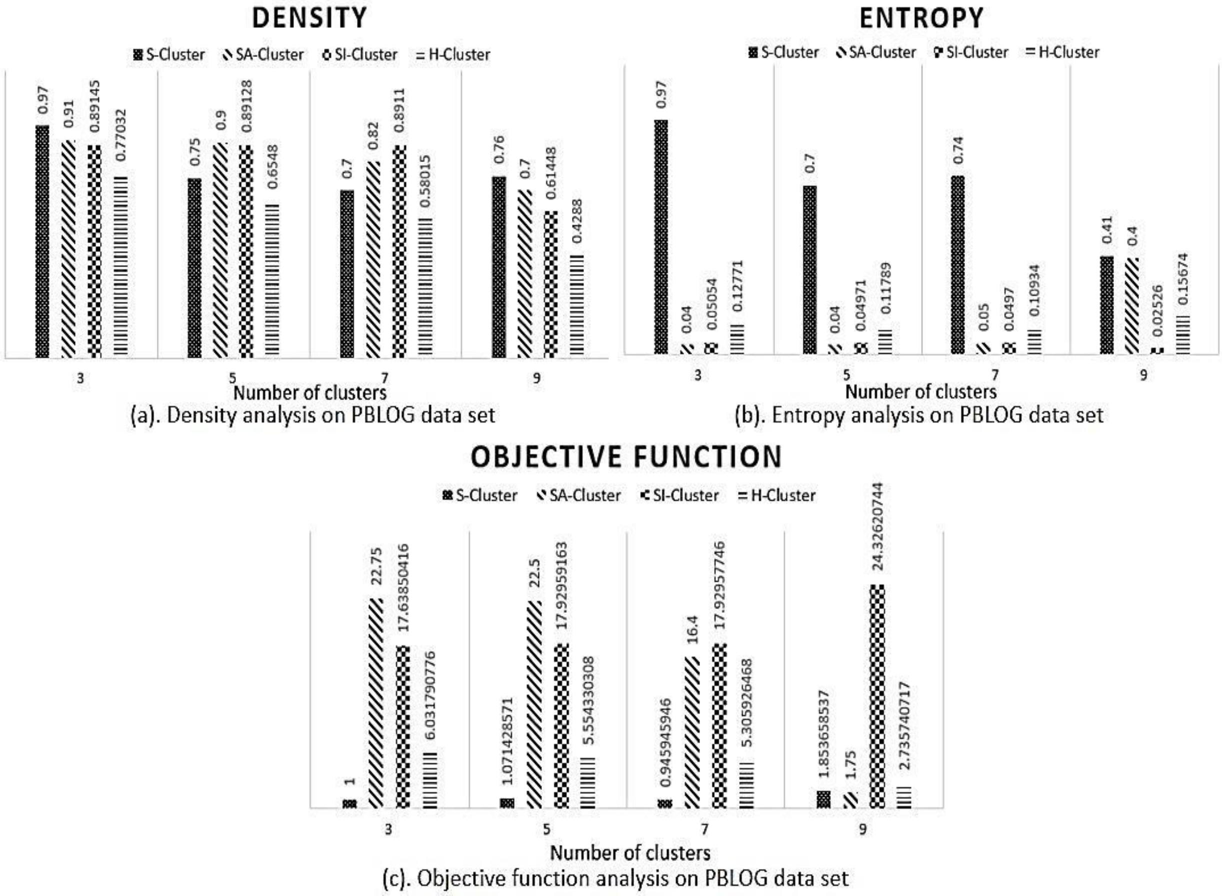


Fig. 9: Alfa's (α) effect on DBLP dataset time complexity.

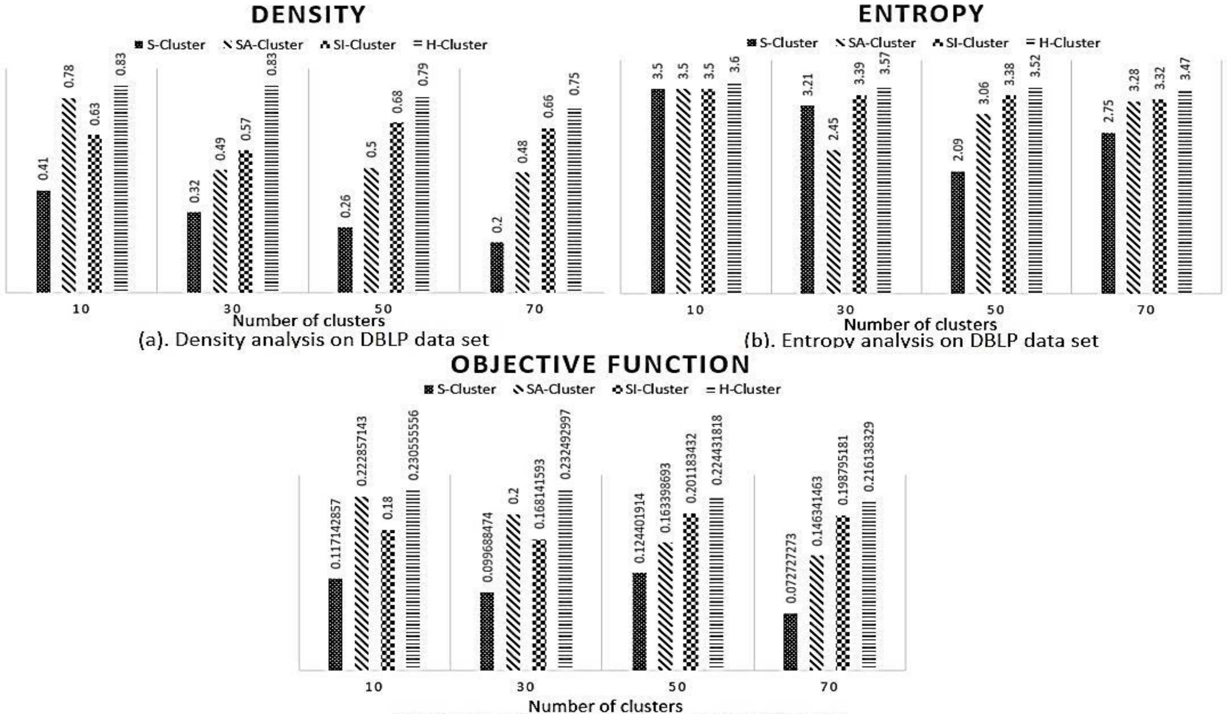


(a). Density analysis on PBLOG data set

(b). Entropy analysis on PBLOG data set

(c). Objective function analysis on PBLOG data set

Fig. 10: Density, entropy and objective function's analysis on *PBLOG data set* ($\lambda = 0.5$, $\alpha = 5\text{ k}$ for PBLOG).



(a). Density analysis on DBLP data set

(b). Entropy analysis on DBLP data set

(c). Objective function analysis on DBLP data set

Fig. 11: Density, entropy and objective function's analysis on DBLP data set ($\lambda = 0.5 \quad \alpha = 40\ \mathrm{k}$ for DBLP).

In average, H-Cluster, as to the (density | entropy | objective function) criteria measurements (-0.1865 | +0.57708 | +3.68919) are improved compared to S-Cluster. S-Cluster algorithm considers only structural aspect, thus, as observed, H-Cluster outperforms S-Cluster as to entropy and objective function criteria.

In general, the obtained results indicate the outperformers of SI-Cluster advantages compared to other known algorithms as to density, entropy and objective function measurement on PBLOG dataset.

The laboratory experiments' results on density and entropy in DBLP dataset are expressed in Fig. 11.

In average, H-Cluster (+0.2375 | -0.4675 | +0.042755) (+0.165 | -0.1425 | +0.038875) outperforms SA-Cluster and SI-Cluster as to (density | entropy | objective function) criteria, respectively. As observed in Fig. 11, though in H-Cluster algorithm considers both the structural and contextual aspects are of concern, it is observed that this algorithm generates more density clusters compared to that of SA-Cluster in all aspects, in addition to, H-Cluster generating more density compared to that of the SI-Cluster algorithm. The SA-Cluster algorithm, similar to H-Cluster and SI-Cluster algorithms, considers both structural and contextual aspects, indicating that H-Cluster outperforms SI-Cluster and SA-Cluster in DBLP dataset. In average, H-Cluster yielded improved (+0.5025 | -0.6525 | +0.122415) on (density | entropy | objective function) when measurements are compared. Though S-Cluster considers only the structural aspect, and, as it observed, this proposed algorithm generates more density clusters for all the issues. In average, SI-Cluster, as to the (density | entropy | objective function) criteria measurements (+0.0725 | -0.325 | +0.003881) (-0.165 | +0.1425 | -0.038875) fails compered to SA-Cluster and H-Cluster, respectively. Except in entropy criteria, failed to SA-Cluster and SI-cluster compared to H-Cluster on entropy criteria. As observed in Fig. 11, considering the fact that both structural and contextual aspects are of concern in SI-Cluster, indicating that SI-Cluster as to DBLP dataset is not as efficient as H-Cluster, the H-Cluster outperforms SI-cluster on DBLP dataset.

In general, the obtained results indicate the outperformers of H-Cluster advantages compared to other known algorithms as to density and objective function measurement on DBLP dataset. Note that small size of PBLOG dataset is one of the main reasons that the proposed algorithm cannot achieve good results.

Conclusion

In this study, an effective method for graph clustering is proposed with the objective of finding cluster with higher density and homogenized nodes as to their features.

The structural and contextual aspects are first, obtained through the given graphs by applying heat diffusion and weighted Jaccard similarities, and next, are integrated into a spatial space. In this space, this newly proposed algorithm seeks to maximize intra-cluster similarity while seeking to reduce the enter-cluster similarity to its lowest level. The clustering results quality is determined through density and entropy criteria. The introduced method outperforms existing algorithms. The empirical studies express the competitive results with respect to the cluster quality and this proposed method operates in polynomial time and is scalable for medium and large scale networks.

Author Contributions

S. Farzi and S. kianian contributed to the design and implementation of the research, to the analysis of the results and to the writing of the manuscript.

Acknowledgment

We thank the editor and all anonymous reviewers.

Conflict of Interest

The author declares that there is no conflict of interests regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancy have been completely observed by the authors.

Abbreviations

PBLOG	Political Weblog network
DBLP	Co-authorship network
KNSAP	k-Node Sammarization attribute pair-wise nodes
H-Cluster	Hierarchical Clustering Algorithm
SI-Clustering	Silhouette index Clustering

References

[1] M.E. Newman, "The structure and function of complex networks," SIAM Review, 45(2):167-256, 2003.

[2] R. Guimera, L.A.N. Amaral, "Functional cartography of complex metabolic networks," Nature, 433: 895, 2005.

[3] E. Ravasz, A.L. Somera, D.A. Mongru, Z.N. Oltvai, A.-L. Barabási, "Hierarchical organization of modularity in metabolic networks," Science, 297: 1551-1555, 2002.

[4] D.M. Wilkinson, B.A. Huberman, "A method for finding communities of related genes," in Proc. the national Academy of sciences, 101: 5241-5248, 2004.

[5] Y. Dourisboure, F. Geraci, M. Pellegrini, "Extraction and classification of dense communities in the web," in Proc. the 16th international conference on World Wide Web: 461-470, 2007.

[6] R. Cazabet, H. Takeda, M. Hamasaki, F. Amblard, "Using dynamic community detection to identify trends in user-generated content," Social Network Analysis and Mining, 2: 361-371, 2012.

[7] K. Konstantinidis, S. Papadopoulos, Y. Kompatsiaris, "Exploring Twitter communication dynamics with evolving community analysis," PeerJ Computer Science, 3: e107, 2017.

[8] C. Bothorel, J.D. Cruz, M. Magnani, B. Micenkova, "Clustering attributed graphs: models, measures and methods," Network Science, 3(3): 408-444, 2015.

[9] H. Cheng, Y. Zhou, J.X. Yu, "Clustering large attributed graphs: A balance between structural and attribute similarities," ACM Transactions on Knowledge Discovery from Data (TKDD), 5(2): 12, 2011.

[10] W. Nawaz, K.-U. Khan, Y.-K. Lee, S. Lee, "Intra graph clustering using collaborative similarity measure," Distributed and Parallel Databases, 33: 583-603, 2015.

[11] S. Farzi, S. Kianian, "A novel clustering algorithm for attributed graphs based on K-medoid algorithm," Journal of Experimental & Theoretical Artificial Intelligence, 30(6): 1-15, 2018.

[12] Z. Xu, Y. Ke, Y. Wang, H. Cheng, J. Cheng, "A model-based approach to attributed graph clustering," in Proc. the 2012 ACM SIGMOD international conference on management of data: 505-516, 2012.

[13] H. Ma, I. King, M.R. Lyu, "Mining web graphs for recommendations," IEEE Transactions on Knowledge and Data Engineering, 24(6): 1051-1064, 2012.

[14] M. Popescu, J. M. Keller, J. A. Mitchell, "Fuzzy measures on the gene ontology for gene product similarity," IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 3(3): 263-274, 2006.

[15] Y. Zhou, H. Cheng, J. X. Yu, "Clustering large attributed graphs: An efficient incremental approach," in Proc. 2010 IEEE 10th International Conference on Data Mining (ICDM): 689-698, 2010.

[16] Y. Tian, R.A. Hankins, J.M. Patel, "Efficient aggregation for graph summarization," in Proc. the 2008 ACM SIGMOD international conference on Management of data: 567-580, 2008.

[17] M.E. Newman, M. Girvan, "Finding and evaluating community structure in networks," Physical review E, 69(2): 026113, 2004.

[18] J. Shi, J. Malik, "Normalized cuts and image segmentation," IEEE Transactions on pattern analysis and machine intelligence, 22(8): 888-905, 2000.

[19] X. Xu, N. Yuruk, Z. Feng, T.A. Schweiger, "Scan: a structural clustering algorithm for networks," in Proc. the 13th ACM SIGKDD international conference on Knowledge discovery and data mining: 824-833, 2007.

[20] Y. Ruan, D. Fuhry, S. Parthasarathy, "Efficient community detection in large networks using content and links," in Proceedings of the 22nd international conference on World Wide Web: 1089-1098, 2013.

[21] S. Kianian, M.R. Khayyambashi, N. Movahhedinia, "Semantic community detection using label propagation algorithm," Journal of Information Science, 42(2): 166-178, 2016.

[22] M. Belkin, P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," Neural computation, 15(6): 1373-1396, 2003.

[23] I.K. RISI, "Diffusion kernels on graphs and other discrete input spaces," in Proc. 19th Int. Conf. Machine Learning, 2002.

[24] J. Lafferty, G. Lebanon, "Diffusion kernels on statistical manifolds," Journal of Machine Learning Research, 6(5): 129-163, 2005.

[25] Y. Li, C. Jia, J. Yu, "A parameter-free community detection method based on centrality and dispersion of nodes in complex networks," Physica A: Statistical Mechanics and its Applications, 438: 321-334, 2015.

[26] S. Ioffe, "Improved consistent sampling, weighted minhash and l1 sketching," in Proc. 2010 IEEE 10th International Conference on Data Mining (ICDM): 246-255, 2010.

[27] L. Kaufman, P. Rousseeuw, Clustering by means of medoids: North-Holland, 1987.

[28] M. Seifikar, F. Saeed, M. Barati, "C-Blondel: An efficient louvain-based dynamic community detection algorithm," IEEE Transactions on Computational Social Systems 7(2): 308-318, 2020.

[29] M. Fozuni. Shirjini, , S. Farzi, A. Nikanjam, "MDPCluster: a swarm-based community detection algorithm in large-scale graphs." Computing, 102: 893-922, 2020.

[30] S.F. Mirmousavi, S. Kianian, "Link Prediction using Network Embedding based on Global Similarity." Journal of Electrical and Computer Engineering Innovations (JECEI), 8(1): 97-108, 2019.

Biographies

Sahar Kianian received her B.Sc. degree in computer Engineering (2007) from razi University, also M.Sc. and Ph.D. degrees in computer Engineering from Isfahan University (2010 and 2016, respectively). She is an assistant professor of computer engineering at Shahid Rajaee University. Her research interests are the application of algorithms, machine learning and data science to complex networks, focuses on protein interactions, connections of neurons and relationships among people. Applications include disease prediction, drug discovery, event detection and tracking, recommendation system, web mining and social influence mining.

Saeed Farzi received the Ph.D. degree in computer engineering from Tehran University, Tehran, Iran, in 2016. He joined the Artificial Intelligence Department, K. N. Toosi University of Technology, Tehran, in 2017. His research interests include machine learning, information retrieval, and social network analysis.

Hamed Samak received his Bachelor and MA from K. N. Toosi university of technology. His research interests are the application of algorithms, machine learning and data science to complex networks, focuses on protein interactions, connections of neurons and relationships among people. Applications include disease prediction, drug discovery, event detection and tracking, recommendation system, web mining and social influence mining

How to cite this paper:

S. Kianian, S. Farzi, H. Samak, “A new clustering algorithm for attributive graphs through information diffusion approaches,” Journal of Electrical and Computer Engineering Innovations, 8(2): 273-284, 2020.

DOI: 10.22061/JECEI.2020.7190.366

URL: http://jecei.sru.ac.ir/article_1472.html