ARTICLE IN PRESS

Swarm and Evolutionary Computation xx (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Regular Paper

A new multi-objective evolutionary framework for community mining in dynamic social networks

Bara'a A. Attea*, Haidar S. Khoder

Department of Computer Science, Baghdad University, Iraq

ARTICLE INFO

Keywords:
Evolutionary clustering
Evolutionary network analysis
Dynamic social networks
Graph partitioning
Social network analysis

ABSTRACT

Evolutionary clustering – clustering in the presence of dynamic shifts of data's topological structure – has recently drawn remarkable attention wherein several algorithms are developed in the study of complex real networks. Despite the growing interests, all of the algorithms are designed based on seemingly the same principle. The primary principle in these evolutionary clustering frameworks is guided by decomposing the problem into two *individual* criteria, *snapshot quality* and *temporal smoothness*. Snapshot quality should properly cluster individuals of a network into interconnected communities. Temporal smoothness, on the other hand, should capture well the dynamic shift of the interconnected clusters from one time step to another. Thus, in the absence of any dynamic behavior, an evolutionary clustering model should be no more than a community detection one in a static network. Unfortunately, all of the developed algorithms are proposed based on discretion of the snapshot quality as a unified of both *intra-* and *inter- connected* community detection model while temporal cost as a community *evolution* detection model. The contribution of this paper starts by noting the limitation of the existing state-of-the-art algorithms. Despite their performance on dynamic complex networks, their formulations lack complete reflection of sufficient community detection model. Our framework, then, models the evolutionary clustering problem by hypothesizing that it should not depart too much from the community detection problem. To support this claim, an alternate decomposition perspective is proposed by projecting the problem, as a multi-objective optimization problem, in the light of *snapshot* and *temporal* of both intra- and inter-community scores. Two snapshot qualities are proposed to individually emphasize the role of intra- and inter- community scores, while temporal cost is proposed to cross-fertilize inter- community score. By applying one of the prominent multi-objective evolutionary algorithms (MOEAs) to solve the proposed multi-objective evolutionary clustering framework and testing it on several synthetic and real-world dynamic networks, we demonstrate the ability of the proposed model to address the problem more accurately than the existing state-of-the-art formulations.

1. Introduction

Due to their practical significance and ever-increasing applicability in many real world dynamic systems, networks and their topological attributes have very recently drawn growing attention and fueled the desire for solving their problems. Examples include online worlds like technological networks, information networks, and social-communication networks such as the internet, World Wide Web, and Facebook. Other interesting examples are biological networks and ecological niches like protein-protein interaction networks and food webs.

Many algorithms have shown up in literature to analyze the behavior of complex networks in a single and, more importantly, in multi time steps. The study of functional homogeneity of group of members (commonly noted as module, co-cluster, or simply, cluster) in the network is much more involved in social network analysis (*SNA*). In its context, a module or a community is a set of individuals with more appearance of intra-connection amongst its members than inter-connection with other communities in the network. Moreover, the aspect of a community can account several types of membership drifting over time resulting in continuous changes in interaction signatures. Thus, by identifying network's communities (and their evolution), several functional phenomena can be depicted and predicted from the network structure. Community mining in evolutionary networks has and continues to have growing applications. Examples include trend analysis in social spheres and dynamic link prediction [1–5].

Capturing the evolution of clusters in dynamic complex networks is first introduced by Chakrabarti et al. [6] and adopted in all state-of-the-

* Corresponding author.
E-mail addresses: baraali@scbaghdad.edu.iq (B.A. Attea), haidarsafeer@yahoo.com (H.S. Khoder).

http://dx.doi.org/10.1016/j.swevo.2016.09.001
Received 16 May 2016; Received in revised form 5 August 2016; Accepted 15 September 2016
Available online xxxx
2210-6502/

Please cite this article as: Attea, B.A, Swarm and Evolutionary Computation (2016), http://dx.doi.org/10.1016/j.swevo.2016.09.001

art approaches (examples include [7–12]). The fundamental issue of evolution of temporal data is addressed in these approaches based on seemingly one common ground and principle inspired primary from the detection of the two participants of the problem: the snapshot patterns and evolutionary patterns of the communities. These two sub-problems are formulated as multi-cost optimization problem content mainly with *snapshot* cost and *temporal* cost. To specify the characteristic of evolutionary clustering problem in these approaches, three design parameters are used, namely, snapshot intra-cluster quality, snapshot inter-cluster quality, and temporal cost. They proved that the interplay of these parameters plays a vital role in the ability of the adopted evolutionary clustering algorithm.

Although all of the existing state-of-the-art frameworks attempt to involve the above mentioned parameters by maximizing snapshot quality of the network at a current time step and minimizing temporal cost of the network between the current time step and the previous one, it allows (as will be demonstrated in our investigations) a certain degree of cross-*competition* between snapshot quality and temporal cost that may become an acute problem while eliminating some promising solutions. This cross-competition, however, can't be overlooked in any evolutionary clustering framework and thus can also act against our framework. Nevertheless, our idea is to lessen the impact of this cross-competition by designing a proper cross-*fertilization* model between the temporal cost and the snapshot inter-cluster connection quality. Once we do that, we can then make a cross-competition between the snapshot intra-cluster connection quality and the designed cross-fertilization model. It is not intended, here, to be an *exact* evolutionary clustering framework, rather, its purpose is to offer a more successful way to maintain the essential characteristic components of evolutionary clustering problem and to explore their combined impacts on the final performance of the model. The remaining sections of this paper present our alternate perspective to solve evolutionary clustering problem in complex networks. The proposed framework should contribute to each of the following two problem solving aspects:

1. How can we characterize the evolutionary clustering problem in dynamic complex networks?
2. How can we shift from the de-facto definition of evolutionary clustering problem and define an alternate and efficient framework to cast and state it?

Starting with Section 2, it gives related backgrounds on the network's evolutionary clustering problem while presenting relevant graph's terminology. State-of-the-art works are then reviewed in Section 3. Our framework is stated and formulated in Sections 4 and 5. Experimental results and corresponding analysis on synthetic and real life social networks are provided in Section 6. Finally, conclusion of the main findings of this paper and further possible ramifications are highlighted in Section 7.

2. Graph clustering and evolutionary clustering

Mathematically, a network is modeled as graph of pairwise edges between its nodes. Assuming, for example, a friendship graph G modeling a social network $\mathcal{N}$, the pairwise friendship connections between individual entities of $\mathcal{N}$ can be modeled by the pair (V, E). The set of n individuals or entities in $\mathcal{N}$ is noted as the set of nodes or vertices $V = \{v_1, v_2, \ldots, v_n\}$ in G while the friendship connection between any pair of individuals in $\mathcal{N}$ is noted as edge (v_i, v_j) in E, i.e. $E = \{(v_i, v_j)|1 \le i, j \le n \wedge i \ne j\}$. Normally, any undirected graph G can be represented by a symmetric square matrix called adjacency or connection matrix A. Rows and columns of A are labeled with the vertices of V and the entry (i, j) is 1 if vertex v_i is adjacent to vertex v_j, i.e. if $(v_i, v_j) \in E$. In list notation, matrix A can be represented by a set of n adjacency lists $L = \{l_1, l_2, \ldots, l_n\}$, one list l_i for each vertex $v_i \in V$ aggregating all 1 entries in row i. Thus, $|l_i| = \sum_{j=1}^{n}(i, j)$ and $|L| = \sum_{i=1}^{n}|l_i|$. Mathematically noted, n is said to be the cardinality of G, $|l_i|$ is said to be the degree of vertex v_i, while $|L|$ is said to be the volume of G.

One of the most important and critical issues addressed in network-based applications is graph co-clustering problem (interchangeably noted as graph bi-partitioning, graph simultaneous clustering, or simply graph clustering problem). Graph bi-partitioning is a fundamental problem in computer science that is proved to be NP-complete [13–15]. Given a graph $G = (V, E)$, the main problem in graph bi-partitioning is to find the set of sub-graphs $G_i = (V_i, E_i) \subset G$ such that the number of inter-edges connecting vertices from two different sub-graphs, usually known as *cut size*, is minimum [16]. Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two sub-graphs of G, the cut set and cut size of G_1 and G_2 are formally defined as in Eqs. (1) and (2), respectively.

$$cut(G_1, G_2) = \{(v_i, v_j) \in E | v_i \in V_1 \wedge v_j \in V_2\} \tag{1}$$

$$|cut(G_1, G_2)| = \sum_{v_i \in V_1 \wedge v_j \in V_2} A(i, j) \tag{2}$$

The second issue that should be carefully addressed in graph bi-partitioning problem is to group individual nodes of the graph into disjoint sets of dense communities. Each community should have *intra*-contributions among its nodes as more as possible than its *inter*-contributions with other communities. Radicchi et al. [17] semantically define a sub-graph $G_i = (V_i, E_i) \subset G$ as a community in a strong sense if for every node v belongs to G_i, the intra-edge connections are larger than inter-connections, i.e. $\forall\, v \in G_i \Rightarrow \sum_{w \in G_i}(v, w) > \sum_{w \notin G_i}(v, w)$. However, if this intra-connections versus inter-connections relation only holds over the aggregation of all G_i's nodes (i.e. if $\sum_{v \in G_i}\sum_{w \in G_i}(v, w) > \sum_{v \in G_i}\sum_{w \notin G_i}(v, w)$), G_i, then, is said to be a community in a weak sense. Inspired by the well known modularity index of Newman and Girvan (normally noted as Q index [18]), several community mining algorithms (e.g. [19–27]) have been suggested in literature to capture the community patterns of complex networks.

Although the graph model of social interactions between individuals has been very successfully used in SNA, it lacks the realization of role of the time at which social interactions occurred. Thus, static friendship graph model can give imprecise information about the evolved patterns of interacted individuals over time. Spiliopoulou [28] presents two main perspectives of community evolution: community tracing and community monitoring. In the light of community tracing, communities can be schematically realized as clusters built at each time step. The analysis of community evolution, then, involves tracing the same community at consecutive time steps and identifying changes. In community monitoring, however, communities can schematically be perceived as smoothly evolving clusters. The analysis of community evolution, then, involves learning models that adapt smoothly from one time step to the next. Recently, several efforts have been directed towards evolutionary network analysis, addressing community mining problem while their intra- and inter- relationships are drifted gradually over time [29].

In evolutionary clustering, a friendship graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ is a graph $\mathbb{G} = G^1, G^2, \ldots, G^T\}$, used to model a dynamic social network $\mathbb{N} = \mathcal{N}^1, \mathcal{N}^2, \ldots, \mathcal{N}^T\}$ across a discrete time steps $1, 2, \ldots, T$. At time step $t|1 \le t \le T$, the static network $\mathcal{N}^t$ speculates the instance $G^t = (V^t, E^t)$ of $\mathbb{G}$ where nodes and edges are added or deleted from $\mathcal{N}^{t-1}$, i.e. $V^t \supsetneq V^{t-1}$ and $E^t \supsetneq E^{t-1}$ (in case of community growth) or $V^t \subsetneq V^{t-1}$ and $E^t \subsetneq E^{t-1}$ (in case of community shrinking), respectively. Theoretically, however, this can be generalized to $\mathcal{N}^t \supseteq \mathcal{N}^{t-1} \supseteq \ldots \supseteq \mathcal{N}^1$ and $\mathcal{N}^t \subseteq \mathcal{N}^{t-1} \subseteq \ldots \subseteq \mathcal{N}^1$, respectively. By instinct, when $\mathcal{N}^t \equiv \mathcal{N}^{t-1} \equiv \ldots \equiv \mathcal{N}^1$, the problem is reduced to the static community clustering problem. Fig. 1 pictorially depicts the evolution of one social networks (being developed by Kim and Han [10]), consisting of 128 nodes. The figure captures the network at the 1st, 2nd, 5th, and 10th time steps.

The main decision arguments of all evolutionary clustering efforts developed in literature are *consistency* and *smoothness* introduced by