Detecting Overlapping Communities of Nodes with Multiple Attributes from Heterogeneous Networks

Kamal Taha[1(✉)] and Paul D. Yoo[2]

[1] Khalifa University, Abu Dhabi, UAE
kamal.taha@ku.ac.ae
[2] Birkbeck College, University of London, London, UK

Abstract. Many methods have been proposed for detecting communities from heterogeneous information networks with general topologies. However, most of these methods can detect communities with homogeneous structures containing nodes with only a single attribute. Investigating methods for detecting communities containing nodes with multiple attributes from heterogeneous information networks with general topologies has been understudied. Such communities are realistic in real-world social structures and exhibits many interesting properties. Towards this, we propose a system called DOMAIN that can detect overlapping communities of nodes with multiple attributes from heterogeneous information networks with general topologies. The framework of DOMAIN focuses on domains (i.e., attributes) that describe human characteristics such as ethnicity, culture, religion, demographic, age, or the like. The ultimate objective of the framework is to detect the *smallest* sub-communities with the *largest* possible number of domains, to which an active user belongs. The smaller a sub-community is, the more specific and granular its interests are. The interests of such a sub-community is the union of the interests and characteristics of the single domain communities, from which it is constructed. We evaluated DOMAIN by comparing it experimentally with three methods. Results revealed marked improvement.

Keywords: Social networks · Heterogeneous information networks · Community detection · Overlapping communities · Multi-domain community

1 Introduction

To be empirically studied, a large number of complex scientific problems need to be depicted as a network representation. Such problems are not limited to specific scientific fields. For example, complex scientific problems in the following fields have been successfully studied after being depicted as a network representation: ecosystems [1, 7, 10, 13, 30], biological systems [4, 5, 6, 16, 23, 25, 26, 33], scientific citations [22], and information systems [3, 8, 14, 15, 19, 29, 32, 34, 35, 38]. However, problems related to social media ecosystem (e.g., Facebook, LinkedIn, Twitter, forums, and blogs) are the most successfully and efficiently studied ones. Detecting community structures is one of the most studied social media ecosystem problems. Each social

X. Wang et al. (Eds.): CollaborateCom 2019, LNICST 292, pp. 760–779, 2019.
https://doi.org/10.1007/978-3-030-30146-0_51

network has a specific community structure. Such structure can be studied for understanding the dynamics of the network. There are numerous reasons for detecting such communities from social networks. For example, there may be a need for classifying the members of some social media network into communities that reflect the organization of the society. A society can be organized into unions that reflect some social criteria. Example of such unions are social groups, colleagues, families, and villages. Such classification is useful in identifying many features that can be used for community membership prediction. It also helps in understanding the dynamics of the members of a community. A cohesive community is defined as a group of densely connected individuals via some common social characteristics such as interests. A "good" community is widely defined as cohesive, compact, and strongly connected internally, but sparsely connected with the remaining parts of the network.

Current methods cluster nodes based on two types of information: network data and attribute data. The network data depicts the relationships between some objects. The attribute data characterizes the objects. These methods employ different techniques that cluster nodes by grouping them either based on their network structural data or attribute data. Most of the methods that cluster nodes based on network structure employ probabilistic generative models to infer the posterior memberships of a community [3, 9, 10, 21, 33, 36]. Most of the methods that cluster nodes based on attribute data can be classified as: (1) methods use the connections between nodes (i.e., link structure) to perform the clustering [3, 11, 13, 28], (2) methods use node attributes to detect the network's communities [4, 37], and (3) methods use both link structure and node attributes to perform clustering [2]. The methods under the first classification overlook the nodes' attributes, which hold important clustering characteristics. The methods under the second classification overlook the important structural relationships between nodes. The methods under the third classification combine the structural and attribute information so that nodes are grouped not only based on the density of their connectivity, but also their common attribute similarities. A large number of these methods detect communities from heterogeneous information networks, which are realistic and exhibits interesting properties. For example, an academic network may include multiple heterogeneous attributes such as author names, journal/conference names, and keywords. However, many of these methods detect communities with only certain topological structures [1, 9, 12–14, 18, 29, 30, 34]. To overcome this, a number of methods have been proposed for detecting communities from heterogeneous information networks with general topologies [20]. However, most of these methods can detect communities with homogeneous structures containing nodes with only a single attribute. That is, they may not detect a community of nodes with multiple attributes. Towards this, we propose in this paper a system called DOMAIN (**D**etecting **O**verlapping **M**ulti-**A**ttributed **I**nformation **N**odes) that can detect overlapping communities of nodes with multiple attributes from heterogeneous information networks with general topologies.

The framework of DOMAIN focuses on attributes that describe human characteristics such as ethnicity, culture, religion, demographic, age, or the like. We use the term "domains" to refer to such attributes. Heterogeneous multi-domain communities are realistic and resemble many real-world communities. For example, a multi-domain community formed from the domains ethnicity, religion, age, and demography can

represent a portion of individuals from a specific ethnic group, who follow a specific religion, who live in a specific neighborhood, and belong to a specific age range. Such a community is realistic in real-world community structures and exhibits many interesting properties. Therefore, DOMAIN aims at detecting the smallest overlapping sub-communities with the largest possible domains, to which active users belong. This is because, the smaller a sub-community is, the more specific and granular its interests are. The interests of such a sub-community is the union of the interests and characteristics of the single domain communities, from which the sub-community is constructed. The main contributions of this paper are summarized as follows:

1. Proposing a methodology for extracting the set of dominant keywords (e.g., buzzwords) from the messages associated with a specific social group to act as a potential representative of the social group.
2. Proposing a graphical model that represents cross-communities and their ontological relationships. The model accounts for all sub-communities with multiple domains that exist due to the interrelations between communities.
3. Proposing a novel and efficient methodology for identifying the smallest sub-communities with the largest number of domains, to which active users belongs.
4. Evaluating our proposed method by comparing it experimentally with three other methods.

2 Concepts Used in the Paper

We call an information network a heterogeneous information network, if the number of attributes and number of links of its nodes are $|N| > 1$ and $|L| > 1$, respectively; otherwise, the information network is a homogeneous information network. We use the term "domain" to refer to a common characterizing attribute of the nodes of a community within a heterogeneous information network. A domain (i.e., a characterizing attribute) defines a community based on a specific and known social group characteristic such as ethnicity, religion, belief, demography, culture, pursuit, area of activity, or the like. We use the term Lone-Domain Community (LDC) to refer to a group of individuals who share a single common domain. For example, individuals, who belong to a specific ethnic group form a LDC. We formalize the concept of LDC in definition 1.

Definition 1 - Lone-Domain Community (LDC): *LDC is an aggregation G of individuals within an information network G = (V, E) with schema (A, R), where each* $x, y \in G\,(x \neq y)$ *share one single common attribute mapping* ∂*:* $V \rightarrow A$ *and link type mapping* ψ*:* $E \rightarrow R$*. That is, a LDC is defined by a common characterizing attribute mapping A, with links as relations from R.*

The smaller a community is, the more specific and granular its interests are. Towards this, we introduce a granular class of communities called Multi-Domain Community (MDC), which is formed from two or more LDCs. Thus, a MDC is a group of individuals who share multiple common domains (e.g., ethnicity, religion, etc.). The size of a MDC is usually smaller than each of the LDCs forming it. An Overlapping Multi-Domain Community (OMDC) is a MDC formed from the intersection of two or

more LDCs with different domains within a heterogeneous information network. That is an OMDC is an aggregation of individuals who share common cross-communities (i.e., inter-communities) domain characteristics. Therefore, an OMDC can be formed from the overlapping of two or more LDCs that belong to different domains.

In general, an OMDC is a granular class of communities formed from two or more LDCs with different domains. As an example, an OMDC can be a portion of individuals who belong to a same ethnic group ETH(x), who also follow a same religion REL(y), and are also descendants from a same national origin ORG(z). Thus, this OMDC is formed from the intersection: $\mathrm{ETH}(x) \cap \mathrm{REL}(y) \cap \mathrm{ORG}(z)$. Intuitively, the characteristics of an OMDC are more granular and specific than the characteristics of each of the LDCs, from which the OMDC is formed. In the framework of DOMAIN, an OMDC is represented by *the set of the overlapped LDCs*, from which the OMDC is formed.

We model OMDCs and their hierarchical relationships using a graphical representation called Overlapping Multi-Domain Communities Graph (OMDCGraph). In an OMDCGraph, each LDC is represented by a node. An OMDC formed from the overlapping of two LDCs C_1 and C_2 is represented by a node $\{C_1, C_2\}$. The ontological relationship between the node $\{C_1, C_2\}$ and each of the nodes C_1 and C_2 is represented by the link connecting them. An OMDCGraph accounts for all the OMDCs that exist due to the interrelations between LDCs of different domains. We formalize the concept of OMDCGraph in Definition 2.

Definition 2 - Overlapping Multi-Domain Communities Graph (OMDCGraph): *An OMDCGraph is a graphical representation of the ontological relationships between cross-communities OMDCs. It consists of a pair of sets (V, E). V is a finite set of nodes depicting LDCs of various domains and the OMDCs formed from the overlapping of these LDCs. E is a set of edges depicting the binary relations on V. An OMDC at a hierarchical level n consists of at least n LDCs. If two OMDCs contain at least one common LDC (i.e., an overlapping LDC), they are linked by an edge to denote their class-subclass relationship. The subclass has its own characteristics while inheriting the characteristics of its parent class. The set of edges E that denotes class-subclass relationships in an OMDCGraph is formalized as follows:*

$E = \{edge(OMDC_i, OMDC_j): OMDC_i, OMDC_j \in V;\ OMDC_i \cap OMDC_j \neq 0;$ *$OMDC_i$ resides at hierarchical level n and $OMDC_j$ resides at hierarchical level n+1 of the OMDCGraph}.*

For the sake of easy reference, we present in Table 1 abbreviations of the concepts proposed in the paper.

Table 1. Abbreviations of the concepts proposed in the paper

Abbreviation	Description
LDC	Lone-Domain Community
MDC	Multi-Domain Community
OMDC	Overlapping Multi-Domain Community
OMDCGraph	Overlapping Multi-Domain Communities Graph

3 Motivation and Outline of the Approach

3.1 Motivation

In real-world setting, there are always new members wishing to join existing and established communities. This requires a methodology for efficiently identifying all existing communities that share the interests of active users. That is, this process requires a methodology for detecting the LDCs, with which the active user shares domains. Each of the different LDCs, to which this user belongs, has the characteristics of a special domain. A sub-community that possesses all these domains, is the most reflective to the characteristics of the user. That is, the smaller a multi-domain community is, the more reflective it is to the characteristics of its members. Therefore, our proposed method in this paper attempts to identify the smallest and most granular multi-domain sub-communities for a user. A granular multi-domain sub-community (i.e., an OMDC) is a subclass of all the LDCs, to which the user belongs. An OMDC is formed from the intersection of two or more LDCs. The characteristics of an OMDC are more granular and specific than the characteristics of each of the LDCs, from which the OMDC is constructed. Intuitively, the size of an OMDC is smaller than each of the LDCs, from which it is constructed.

Identifying the OMDC, to which a user belongs, requires a method that can detect communities of nodes with multiple attributes from heterogeneous information networks with general topologies. Investigating such methods has been understudied. Most such existing methods can detect communities with homogeneous structures containing nodes with only a single attribute. That is, most these methods cannot detect a community of nodes with multiple attributes. Towards this, we introduce our proposed system DOMAIN. The system focuses on characterizing attributes (i.e., domains) that describe human characteristics such as ethnicity, culture, religion, demographic, age, or the like. The ultimate objective of DOMAIN is to detect the smallest sub-communities with the largest possible domains, to which active users belong.

3.2 Outline of the Approach

The following are the sequential processing steps taken by DOMAIN for detecting the smallest sub-communities with the largest possible number of domains, to which an active user belongs:

(1) *Constructing a Training OMDCGraph (Sect. 4):*
   (a) Collect messages from explicitly declared LDCs of different domains to be used as training datasets. Each set of messages associated with a specific LDC will be used by the system as a training dataset with respect to the LDC.
   (b) Extract a set of *candidate* keywords (e.g., buzzwords) from the messages associated with each LDC to act as a *potential* representative of the LDC.
   (c) Filter the candidate keywords of each LDC to identify the *dominant* ones (the ones that have frequent occurrences in a significant number of messages associated with the LDC). The identified set of dominant keywords will be used by the system as a representative of the LDC.

(d) Construct the training OMDCGraph as follows:
   i. *Construct the root level of the graph:* Each LDC with a unique domain is represented by a node at the root level of the graph.
   ii. *Construct level 1 of the graph*: If there is a significant number of *common* dominant keywords associated with a set S of root LDC nodes, the set converges at level 1 of the graph. The convergence node represents an OMDC, which is represented by the set S.
   iii. *Construct the remaining levels of the graph:* Each combination of OMDC nodes located at level n of the graph converge at level $n + 1$ to form a new OMDC node, if: (1) there exist at least one common LDC in all the OMDC nodes in the combination, and (2) the combination does not include more than one LDC with the same domain. The convergence node is represented by the set of all the LDCs in the combination. This process continues until no new OMDC can be formed at a new level.

(2) *Identifying the LDCs to which an active user belongs (Sect. 5):*
   (a) Extract the dominant keywords from the messages associated with the active user using the same techniques described in steps 1-b and 1-c.
   (b) If the active user and a root LDC node have significant common dominant keywords associated with their messages, the active user belongs to this LDC.

3) *Identifying the smallest OMDC with the largest number of domains to which the active user belongs (Sect. 6)*:
   (a) Mark the root nodes identified in step 2-b.
   (b) The active user's smallest OMDC is located at the convergence of the *longest paths* originated from the marked nodes described in step 3-a. The active user belongs to all the LDCs comprising this convergence OMDC.

4 Constructing a Training OMDCGraph

4.1 Extracting the Dominant Keywords from the Training Messages Dataset Associated with a LDC

After extracting the set of candidate keywords (e.g., buzzwords) from the messages associated with a LDC, DOMAIN filters these keywords to keep only the dominant ones (the ones that have frequent occurrences in significant number of the messages associated with the LDC). This is because a keyword is uninformative, if it occurs in a few messages or/and it has few occurrences in the messages. To overcome this, DOMAIN keeps only the candidate keywords that have frequent occurrences in a significant number of messages associated with the LDC. This set of identified dominant keywords will be used by DOMAIN as a representative of the LDC.

DOMAIN identifies the dominant keywords by associating each keyword with a score that reflects its dominance status with regards to the other keywords. It assigns a pairwise beats and looses indicator for each candidate keyword that occur in the

messages associated with the LDC. A beat-loose table is constructed as follows. The entries of the table are (k_i, k_j) where k_i denotes keyword i while k_j denotes keyword j.

Let n_i be the number of times that the number of mentioning of k_i in the messages associated with the LDC is greater than that of k_j. Let n_j be the number of times that the number of mentioning of k_j in the messages is greater than that of k_i. If $ni > n_j$, entry (k_i, k_j) will assigned the indicator symbol "+". Otherwise, it will be assigned the indicator "−". If $n_i = n_j$, entry (k_i, k_j) will assigned the indicator symbol "0". We now formalize the concept of pairwise score of a keyword in Definition 3:

Definition 3 – Pairwise score of a keyword: *Let the denotation $k_i > k_j$ means that the number of times the number of mentioning of k_j in the messages associated with a LDC is greater than that of k_i. The pairwise score of the keyword k_i equals: $\left|\{k_j \in K_{LDC} : k_i > k_j\}\right| - \left|\{k_j \in K_{LDC} : k_j > k_i\}\right|$, where K_{LDC} denotes the set of all candidate keywords in the messages associated with the LDC.*

Finally, the keyword k_i will be given a dominance score S, which is computed as follows. Let N_b be the number of times that k_i *beat* all other keywords. Let N_l be the number of times that k_i *lost* to all other keywords. The dominance score of k_i (S_{ki}) equals: $S_{ki} = N_b - N_l$. The summation of the dominance scores of all keywords is zero. If m is the number of keywords, the highest possible dominance score is $(m - 1)$, while the lowest possible dominance score is $-(m - 1)$. Each keyword is given a normalized dominance score $\bar{S}$. The keywords, whose normalized dominance scores are greater than a threshold β are considered dominant. The rest of the keywords are excluded and considered uninformative. As shown in Eq. 1, β is the value that is less than the mean of the normalized dominance score by the standard error of the mean.

$$\beta = \frac{1 - \sqrt{\sum_{\forall k_j \in K_{LDC}} \left(\bar{S}_{k_j} - \frac{1}{|K_{LDC}|}\right)^2}}{|K_{LDC}|} \tag{1}$$

Example 1: To illustrate the process of identifying the dominant keywords, we present a simplistic hypothetical example of 10 keywords that co-occur in 3 messages as shown in Table 2. Table 3 shows the pairwise score S and dominance score $\bar{S}$ of each keyword computed based on the number of occurrences of the keyword in the 3 messages shown in Table 2.

Table 2. Distribution of the 10 hypothetical keywords in the 3 messages shown in Example 1

	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9	k_{10}
m_1	2	4	0	0	3	0	1	2	4	1
m_2	0	2	2	3	4	2	0	0	1	3
m_3	3	5	1	4	5	2	3	1	2	0

Table 3. The pairwise scores of the 10 keywords presented in Table 2 and described in Example 1 computed based on their *beats* and *looses* indicators. The table shows also the dominance scores and normalized dominance scores of the 10 keywords.

Keyword	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9	k_{10}
k_1	0	+	–	+	+	–	–	–	+	–
k_2	–	0	–	–	0	–	–	–	–	–
k_3	+	+	0	+	+	+	+	0	+	+
k_4	–	+	–	0	+	–	–	–	–	0
k_5	–	0	–	–	0	+	–	–	–	–
k_6	+	+	–	+	–	0	+	–	0	+
k_7	+	+	–	+	+	–	0	0	+	0
k_8	+	+	0	+	+	+	0	0	+	–
k_9	–	+	–	+	+	0	–	–	0	–
k_{10}	+	+	–	0	+	–	0	+	+	0
S	**+1**	**+8**	**–8**	**+4**	**+6**	**–2**	**–3**	**–5**	**+2**	**–3**
$\bar{S}$	**0.11**	**0.2**	**0**	**0.15**	**0.17**	**0.08**	**0.06**	**0.04**	**0.13**	**0.06**

4.2 Constructing the Training OMDGraph

In this section, we describe the process of constructing a graphical representation of the ontological relationships between the training OMDCs. That is, we describe the process of constructing an Overlapping Multi-Domain Social Graph (OMDCGraph) that depicts the ontological relationships between the training OMDCs. Each LDC with a unique domain is represented by a node and placed at the root level of the OMDC-Graph. This node itself is represented by the dominant keywords (recall Sect. 4.1) in the messages associated with the LDC.

Level 1 of the OMDCGraph is constructed as follows. The paths originating from a subset S of the set of root nodes converge at level 1 of the graph to form a new OMDC node, if: the frequency of messages associated with each LDC node $N \in S$ that have occurrences of dominant keywords found in the messages associated with each other LDC node $N' \in S$ is significant. The new convergence OMDC node is represented by the set S of nodes. This node inherits the characteristics of each of the LDCs in the set S. Let $F_{N_i}^{N_j}$ be the frequency of messages associated with node N_j that contain occurrences of dominant keywords found in the messages associated with node N_i. $F_{N_i}^{N_j}$ is considered significant, if it is greater than β', which is a heuristically determined threshold. In the framework of DOMAIN, one of the following two frequency formulas is used based on application-specific requirements:

(1) Let M_k be the number of messages containing occurrences of keyword k. The following formula is preferred, if we want to diminish the impact of rare events. That is, if we do not want to consider the occurrences of k for which $M_k = 1$ as twice significant as the occurrence of k for which $M_k = 2$:

$$F_{N_i}^{N_j} = \log\left(1 + \frac{|M^{N_j}|}{|M_{N_i}^{N_j}|}\right)$$

- M^{N_j}: The set of messages associated with node N_j.
- $M_{N_i}^{N_j}$: The set of messages associated with node N_j that contain occurrences of dominant keywords in the messages associated with node N_i.

(2) The following formula is preferred, if the sizes of messages are relatively close. Specifically, it is preferred, if we want to disregard the *size* of messages containing common dominant keywords relative to the overall size of messages:

$$F_{N_i}^{N_j} = \log\left(1 + \frac{MAX|K_{N_i}^{N_j}|}{|M_{N_i}^{N_j}|}\right)$$

- $MAX|K_{N_i}^{N_j}|$: Maximum number of occurrences of the dominant keywords in the messages associated with node N_j in a message associated with node N_i.

The remaining levels of the OMDCGraph are constructed as follows. All unique combinations of OMDC nodes located at level n of the graph are enumerated. Let $\acute{S}$ be the set of all these different combinations at level n. Each subset $\acute{s} \subseteq \acute{S}$ converge at level $n + 1$ to form a new OMDC node, if: (1) there exist at least one common LDC in all the OMDC nodes $\in \acute{s}$, and (2) $\acute{s}$ does not include two or more LDCs with the same domain. The convergence OMDC node is represented by the set of LDCs in $\acute{s}$. The node inherits the characteristics of each LDC in $\acute{s}$. This process concludes when there is no new OMDC node can be formed at a new level. An OMDCGraph accounts for *all* the OMDCs that exist due to the interrelations between LDCs of different domains.

Example 2: From the messages that belong to some social media, consider that DOMAIN identified the seven LDCs shown in Table 4, which fall under four different domains. Consider that DOMAIN constructed the OMDCGraph shown in Fig. 1 using the techniques described in Sect. 4. Each OMDC node at level 2 of the graph is formed from the convergence of two OMDC nodes at level 1 that have at least one common LDC and do not have two or more LDCs with the same domain. For example, the OMDC node {REL(y), ETH(x), NBHD(y)} at level 2 resulted from the convergence of the following two OMDC nodes at level 1, which include the common LDC node ETH (x) and do not include more than one LDC with the same domain: {ETH(x), NBHD(y)} and {ETH(x), REL(y)}. This convergence node {REL(y), ETH(x), NBHD(y)} is represented by the set of LDCs forming it and it denotes the portion of individuals who follow the same religion REL(y), who also belong to the same ethnic group ETH(x), and who also live in neighbourhood NBHD(y).

Table 4. Seven hypothetical LDCs with four different domains used in the construction of the OMDCGraph in Fig. 1 and described in Example 2

LDC	Domain	Description
NBHD(x), NBHD(y)	Neighbourhood-based	Users who live in neighbourhood NBHD(x) and NBHD(y)
REL(x), REL(y)	Religion-based	Users who follow religions REL(x) and REL(y)
ETH(x), ETH(y)	Ethnicity-based	Users who are descendants of ethnicities ETH(x) and ETH(y)
ORG(x)	Region-based	Users from national origin (ORG(x))

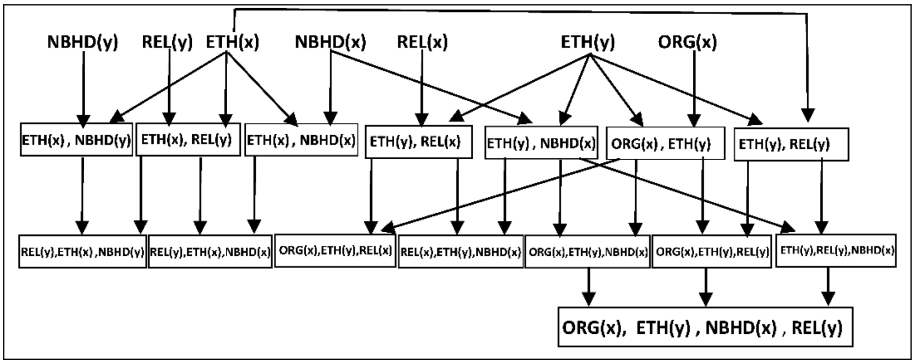

Fig. 1. A training OMDCGraph constructed based on the information described in Example 2

5 Identifying the LDCs to Which an Active User Belongs

As described in Sect. 4.2, each LDC with a unique domain is represented by a node and placed at the root level of the training OMDCGraph. As described in Sect. 4.1, each of these LDC nodes is represented by the set of dominant keywords extracted from the messages associated with it. From the set of LDC nodes, OMDCGraph identifies the subset, to which as active user belongs. It determines that the active user belongs to a LDC, if the messages associated with the user contains significant number of keywords, whose *ontological concepts* fall under the dominant keywords of the LDC. An ontology describes the concepts in a domain of discourse. Let k "kind of" k' means that class k is a subclass of class k' in an ontology. k' is the highest general superclass of k in a defined ontology hierarchy. k shares the same domain, cognitive characteristics, and properties of k'. DOMAIN labels all nodes in an ontology with the label of the root node.

First, DOMAIN fetches the user's messages for keywords, whose ontological concepts fall under the ontological concepts of the dominant keywords of each root LDC node in the OMDCGraph. Consider that d_i is one of these nodes. Each of the dominant keywords representing d_i is considered a root ontology. Then, DOMAIN fetches the user's messages for keywords, whose ontological concepts fall under the

root ontologies of d_i. That is, it fetches the user's messages for keywords that fall under each of the ontological concept of the dominant keywords of d_i. Consider that the word "entertainer" is one of the dominant keywords of d_i.

The user is considered to belong to the LDC represented by the node d_i, if the number of keywords in the user's messages that fall under the ontological concept of d_i is greater than a heuristically determined threshold. However, some of the keywords in the user's messages that fall under the ontological concepts of the dominant keywords of d_i may not be reflective of the community represented by d_i. This happens when these keywords are associated with many *other* LDC root nodes. To overcome this, the DOMAIN considers a keyword as reflective of d_i if the probability of its occurrences in the messages associated with d_i is statistically significantly different from the probability of its occurrences in messages associated with all other LDC root nodes. Towards this, DOMAIN uses the Z-Score statistical test to filter the keywords in the user's messages that fall under the ontological concept of d_i and keeps only the ones that are better reflective of d_i. This is done by calculating the differences of the occurrence probabilities of the keywords across the different community nodes.

The Z-score "$Z - score_k^N$" of a keyword k extracted from the messages associated with the LDC node N in the OMDCGraph is computed as in Eq. 2. In the framework of our DOMAIN, the keyword k is considered a reflective of the characteristics of N, if $Z - score_k^N >$ "-1.96" standard deviation, using a 95% confidence level.

$$Z - score_k^N = \frac{\left(\left|M_k^N\right| / \left|M^N\right|\right) - \left(\left|M_k^{N'}\right| / \left|M^{N'}\right|\right)}{\sigma} \tag{2}$$

- M_k^N: Set of messages associated with LDC node N that contain occurrences of the keyword k.
- $M_k^{N'}$: Set of messages associated with all other LDC nodes N' that contain occurrences of the keyword k.
- M^N: Set of messages associated with LDC node N.
- $M^{N'}$: Set of messages associated with all other LDC nodes N'.
- σ: population's standard deviation.

DOMAIN is built on top of Stanford CoreNLP [12] and Protégé [24]. DOMAIN uses Stanford CoreNLP for generating keyword lemmas and recognizing named entities in the messages associated with the user. It uses Protégé for ontology alignment and the matching between the keyword lemmas in the user's messages and the dominant keywords in the training dataset (i.e., the dominant keywords representing the root nodes in the training OMDCGraph). That is, DOMAIN uses Protégé for capturing the correspondences between the keywords in the user's messages and the training dominant keywords. Ontology matching (i.e., ontology alignment) is the procedure of identifying the correspondences between different concepts. DOMAIN through Protégé checks if there is a match between a dominant keyword and a keyword (or its respective ontological sub-categories) extracted from the user's messages.

6 Identifying the Smallest OMDC with the Largest Number of Domains to Which an Active User Belongs

To identify the smallest OMDC with the largest number of domains to which an active user belongs, DOMAIN performs the following:

(1) It marks the *root* LDC nodes in the OMDCGraph, to which the active user belongs (recall Sect. 5 for how OMDCGraph identifies these root LDC nodes).
(2) It traverses through the paths of the OMDCGraph starting from the marked root LDC nodes to identify the OMDC nodes, at which *all* the paths convergence at each level of the graph. That is, by navigating the paths originating from the marked nodes, OMDCGraph identifies the OMDC nodes located at the convergences of *all* the paths at each level.
(3) From among the different OMDCs identified in step 2, the smallest OMDC with the largest number of domains, to which the active user belongs, is the one located at the convergence of all the *longest* paths originating from the marked root nodes. That is, this OMDC node is positioned at the intersection of *all longest paths* originating from the marked root nodes.

If all longest paths originated from n root nodes, the user's smallest OMDC located at the convergence of these paths is usually formed from m LDCs, where $m > n$. That is, if DOMAIN identified n *explicit* LDC root nodes for the user (using the techniques described in Sect. 5), the user's smallest OMDC with the largest number of domains is likely to contain greater than n LDCs. The extra LDCs (i.e., the $m - n$ LDCs) are identified *implicitly* by DOMAIN based on the structure of the OMDCGraph and the interrelations between the different OMDC nodes. The user's messages may not contain keywords that directly refer to these extra LDCs.

Example 3. Consider that DOMAIN traversed the paths of the OMDCGraph shown in Fig. 1 and described in Example 2 in order to identify the smallest OMDC with the largest number of domains, to which an active user belongs. Using the techniques described in Sect. 5, consider that DOMAIN identified the following explicit LDC nodes, to which the active user belongs: (1) neighborhood NBHD(x), and (2) national origin ORG(x). First, DOMAIN would mark the root LDC nodes NBHD(x) and ORG (x) as shown in the OMDCGraph in Fig. 2. Then, starting from the marked two root nodes, DOMAIN would navigate through the paths to identify the convergence OMDC nodes as shown in Fig. 2. For easy reference, the path originating from NBHD(x) is marked with dotted red and the path originating from ORG(x) is marked with dashed blue. There can be several convergence OMDC nodes at different levels of the graph, but there is only one convergence node in this particular example. As Fig. 2 shows, the smallest OMDC with the largest number of domains, to which the active user belongs is the following:

$$\{\text{ORG}(x), \text{ETH}(y), \text{NBHD}(x), \text{REL}(y)\}$$

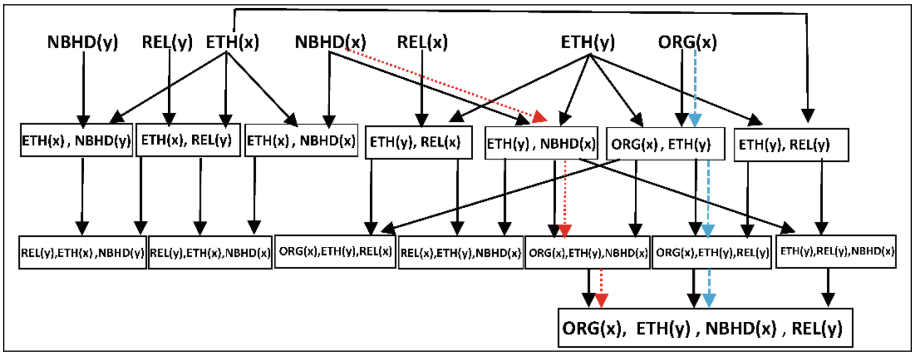

Fig. 2. The convergence of the longest paths originating from the root nodes in the OMDCGraph that indicates the smallest OMDC, to which the user described in Example 3 belongs.

This OMDC is located at the convergence of the longest paths originating from the root nodes NBHD(x) and ORG(x). The following observations can be drawn from the result:

- The paths originated from only *two* root LDC nodes and that the active user's smallest OMDC node contains *four* LDC nodes. That is, from the user's two *explicitly* identified LDCs, DOMAIN could identify the user's smallest OMDC, which contains four LDCs. Two of them are *implicitly* inferred by the system.
- The two extra implicitly identified LDCs (i.e., ETH(y) and REL(y)) are inferred by DOMAIN based on the structure of the OMDCGraph and the interrelations between the different OMDCs.

Every time DOMAIN identifies the smallest OMDC for a user, it will enhance the training dataset and OMDCGraph accordingly. Let N be the smallest OMDC node identified by DOMAIN for an active user u. DOMAIN will enhance the training dataset by updating it as follows. It will add the list of messages associated with u to the list of messages associated with each LDC node $N' \in N$. That is, the list of training messages associated with each N' will be incremented by including the list of messages associated with u. Accordingly, DOMAIN will update and optimize the following: (1) the number of keywords' occurrences in the messages associated with each $N' \in N$ (e.g., recall Table 2), and (2) the pairwise score S and dominance score $\bar{S}$ (recall Table 3) of each keyword in the messages associated with each $N' \in N$. For the sake of conserving computation time, we advocate updating the pairwise score S and dominance score $\bar{S}$ only at certain intervals (i.e., update points). That is, the update is based on *all* OMDCs identified between intervals and not based on each one of them individually.

7 Experimental Results

We implemented DOMAIN in Java. We ran the system under Windows 10 Pro using Intel(R) Core(TM) i7-6820HQ processor. The machine has 2.70 GHz CPU and 16 GB RAM. We evaluated DOMAIN by comparing it experimentally with Sharma et al. [27]. Sharma et al. [27] uses the concept of group accretion, which is the process of increasing the size of a group by adding new more members. It uses the communication paths in a network to measure the degree of relationships between a group and a person outside the group. Given a group with n members, [27] predicts the likelihood of a new member outside the group for being absorbed in the group, where the size of the group will be incremented to $n + 1$. The authors proposed three different methods inspired by dyadic link prediction (DLP) techniques and sociology theories. Each of these methods assigns a score to each group to reflect its similarity (i.e., affinity) with the person outside the group. The first method is called GKS. It extends the Katz method [17], which enumerates network paths. GKS makes predication by employing a DLP-inspired unsupervised path counting. The second method is called BRWS. It uses a semi-supervised learning approach inspired by network alignment algorithms. It identifies each cycle that passes through each group and the remaining groups. The third method is called GLPS. It employs a semi-supervised method inspired by hypergraph label propagation techniques. We evaluated and compared the accuracy of communities detected by DOMAIN, GKS, BRWS, and GLPS in terms of F1-score and Adjusted Rand Index (ARI), with reference to a ground-truth dataset. F1-score is the harmonic average of precision and recall, and is computed as shown below:

$$\text{F1} - \text{socre} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

ARI computes the expected similarity of all pair-wise comparisons between two clusters (e.g., between a ground-truth community and a community detected by a method), as shown in the formula below:

$$\text{ARI} = \frac{\text{index} - \text{expected index}}{\text{maximum index} - \text{expected index}}$$

- $\text{Index} = \sum_{ij} \binom{n_{ij}}{2}$
- $\text{Expected index} = \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}$
- $\text{Max index} = \frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right]$

We evaluated and compared DOMAIN, GKS, GLPS, and BRWS using the DBLP dataset [31]. We adopted the same experimental setup and the same dataset used for evaluating GKS, GLPS, and BRWS as described in [27]. We used the following same

experimental setup described in [27]: (1) the same training and test periods of main splits, (2) the same metrics, and (3) the same DBLP dataset. Below are brief descriptions of the mentioned DBLP, split periods, and metrics:

- The DBLP dataset [31] was extracted from publications in 22 different computer science subfields from 1930 to 2011.
- The dataset was divided to different splits as shown in Table 5. As shown in the table, each split is marked with a fixed end year of the training dataset. Papers published between the years 2004 and 2007 are used for the training while papers published between the years 2008 and 2010 are used for the testing.
- The metrics used for the evaluation are defined as follows:

$$\text{Precision@}\,N_{top}\,(\text{IA}) = \frac{\text{Number of groups correctly predicted using IA process from top} - N_{top}\ \text{list}}{N_{top}}$$

$$\text{Recall@}N_{top}\,(\text{IA}) = \frac{\text{Number of collaborations correctly predicted using IA process from top} - N_{top}\ \text{list}}{\#\ \text{of actual IA generated groups}}$$

- IA: Incremental accretion.
- N_{top}: The top sorted N unique set of IA.
- *Top-*N_{top}: The highest scoring in the sorted N unique set of IA.

As Table 5 shows, we divided the dataset into the same training and test periods (splits) as described in [27]. These divisions are the same ones used by Sharma et al. [27] in evaluating GKS, BRWS, and GLPS. Table 6 shows the prediction accuracy of the methods based on the divisions of the dataset shown in Table 5, using the per-group metrics Precision@N_{top}(IA) and Recall@N_{top}(IA) for N_{top} = 100 as described in [27]. The values shown in Table 6 for GKS, BRWS, and GLPS are the same ones listed in [27].

Table 5. Dividing the dataset into the same training and test periods (splits) as described in [27]

Boundary Yr	Split No.	Train	Test
2007	Main Split	2004–2007	2008–2010

Table 6. The prediction accuracy of the methods using the per-group metrics Precision@N_{top}(IA) and Recall@N_{top}(IA) for N_{top} = 100 as described in [27]. The values shown in the table for GKS, BRWS, and GLPS are the same ones listed for these methods in [27].

	GKS	GLPS	BRWS	DOMAIN
AvgPrecision@100(IA)	0.0210	0.0349	0.0355	0.147
AvgRecall@100(IA)	0.3176	0.6034	0.6050	0.6083

We also compared the accuracy of the four methods for detecting the DBLP communities in terms of F1-score (Fig. 3) and ARI (Fig. 4).

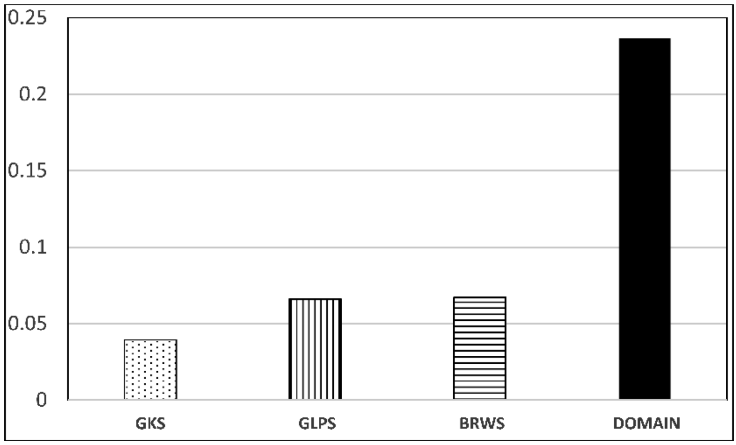

Fig. 3. The accuracy of each method for detecting the DBLP communities in terms of F1-score

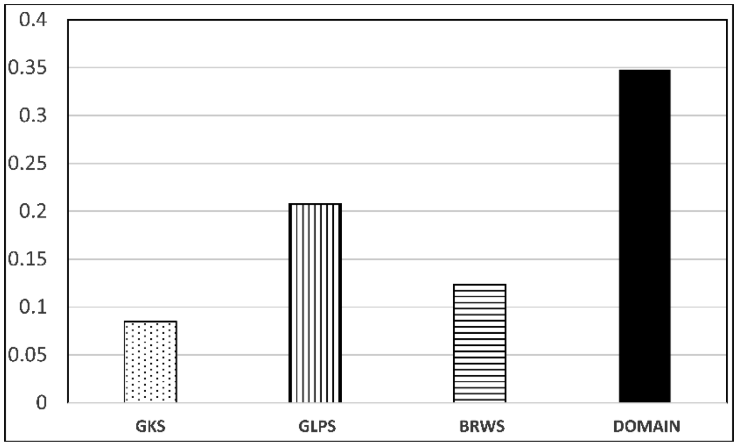

Fig. 4. The accuracy of each method for detecting the DBLP communities in terms of ARI

8 Discussion of the Results

As Table 6 and Figs. 3 and 4 show, DOMAIN outperformed the GKS, BRWS, and GLPS methods in terms of AvgPrecision@100(IA), AvgRecall@100(IA), F1-score, and ARI. We attribute the performance of DOMAIN over the other three methods to its good predictive capabilities and also the limitations of these three methods. In general, the GKS method did not perform well, while the GLPS method performed well compared to the BRWS and GKS methods. Based on our observations of the

experimental results, we attribute the poor performance of the GKS method to several limitations, mostly related to Katz score employed by the method. We attribute the relative performance of the GLPS method over BRWS method to the fact that the later considered the paths and cycles over the network of groups while the former did not.

In general, the experimental results revealed that DOMAIN detected with marked accuracy communities of nodes with multiple attributes from heterogeneous information networks with general topologies. We attribute this, mainly, to the graphical representation modelling (i.e., OMDCGraph) employed by DOMAIN, which represents the ontological relationships between *all* cross-communities. This is because the modelling techniques adopted in OMDCGraph account for all the multi-attribute communities with different domains that exist due to the interrelations between communities. The experimental results showed also that DOMAIN's detection accuracy increases as the number of attributes in a detected overlapped community increases. On the other hand, the results showed that the number of attributes in a detected overlapped community is irrelevant to the detection accuracy of the other three methods.

To better demonstrate the impact of a community's number of attributes on its detection accuracy by each method, we performed the following. We classified the communities detected by each method into sets based on the number of attributes in the communities. For each of the four methods, each set includes the communities detected by the method that have the same number of attributes. Then, we computed the overall average F1-score for each set. Figure 5 shows the results. As the figure shows, the detection accuracy of DOMAIN increases constantly as the number of attributes in a community increases. We attribute this to the capability of DOMAIN to detect the smallest sub-communities with the largest possible domains, to which users belong. This is because, the smaller a community is, the more specific and granular its interests are, which is evident in the dataset used in our experiments. These interests are included in the profiles of users in the dataset.

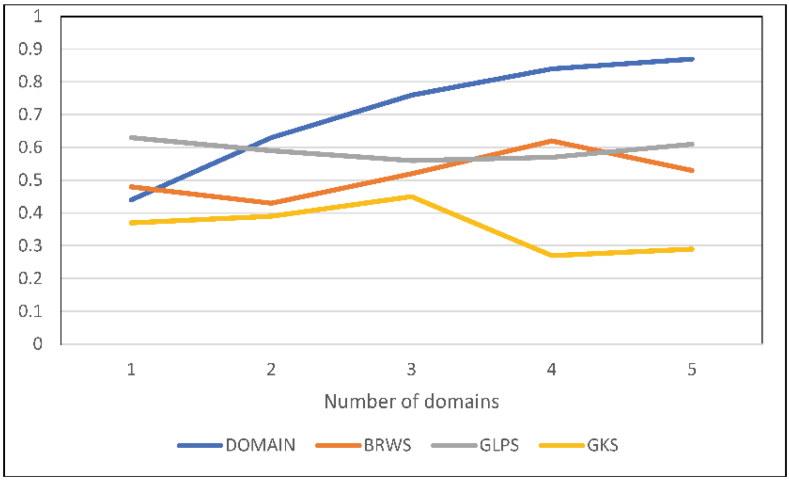

Fig. 5. The overall average F1-score for each set of detected communities that have the same number of attributes.

9 Conclusion

We proposed in this paper a system called DOMAIN that can detect communities of nodes with *multiple attributes* from heterogeneous information networks with general topologies. The framework of DOMAIN focuses on attributes (i.e., domains) that describe human characteristics such as ethnicity, culture, religion, demographic, or the like. Detecting such heterogeneous multi-domain sub-communities is crucial for understanding and analysing the structures and dynamicity of real-world social networks.

DOMAIN aims at detecting the smallest OMDC with the largest possible number of domains, to which an active user belongs. The smaller a sub-community is, the more specific and granular its interests are. The interests and characteristics of such an OMDC is the union of the interests and characteristics of the LDCs, from which it is constructed. DOMAIN identifies the user's smallest OMDC with the largest number of attributes as follows. It models training OMDCs using a graphical representation called OMDCGraph, which represents the ontological relationships between the OMDCs. In the graph, each LDC is represented by a node at the root level. The paths from some root nodes converge at level 1 of the graph to form a new OMDC node, if the frequency of messages associated with these nodes that contain common dominant keywords is significant. Each OMDC node at level n+1 of the graph is formed from the convergence of two or more OMDC nodes at level n that have at least one common LDC and do not have two or more LDCs with the same domain. The user's smallest OMDC with the largest number of domains is located at the convergence of the longest paths originating from root nodes representing LDCs that have significant matches with the user.

We evaluated DOMAIN by comparing it experimentally with the three methods proposed by Sharma et al. [27]. The experimental results showed that DOMAIN outperformed the three methods in terms of AvgPrecision@100(IA), AvgRecall@100 (IA), F1-score, and ARI. The results showed that DOMAIN's accuracy increases as the number of attributes in an overlapped detected community increases. We attribute this to the strong graphical representation modelling (i.e., OMDCGraph) employed by DOMAIN. This is because OMDCGraph accounts for all cross-communities with different domains that exist due to the interrelations between communities. However, the results showed that DOMAIN achieves modest results when the percentage of incomplete users' profiles in a detected community is rather large. We will investigate this shortcoming in a future work.

References

1. Aggarwal, C., Xie, Y., Yu, P.: Towards community detection in locally heterogeneous networks. In: SDM, pp. 391–402 (2011)
2. Akoglu, L., Tong, H., Meeder, B., Faloutsos, C.: PICS: parameter-free identification of cohesive subgoups in large attributed graphs. In: Proceedings of the SIAM International Conference on Data Mining, 2012, USA, pp. 439–450 (2012)

3. Al Zaabi, M., Taha, K., Martin, T.: CISRI: a crime investigation system using the relative importance of information spreaders in networks depicting criminals communications. IEEE Trans. Inf. Forensics Secur. **10**(10), 2196–2211 (2015)
4. Al-Aamri, A., Taha, K., Homouz, D., Al-Hammadi, Y., Maalouf, M.: Analyzing a co-occurrence gene-interaction network to identify disease-gene association. BMC Bioinformatics **20**, 70 (2019)
5. Al-Aamri, A., Taha, K., Homouz, D., Al-Hammadi, Y., Maalouf, M.: Constructing genetic networks using biomedical literature and rare event classification. Sci. Rep. **7**, 15784 (2017)
6. Al-Jarrah, O., Yoo, P., Taha, K., Muhaidat, S.: Randomized subspace learning for proline cis-trans isomerization prediction. IEEE/ACM Trans. Comput. Biol. Bioinform. **12**(4), 763–769 (2015)
7. Boden, B., Ester, M., Seidl, T.: Density-based subspace clustering in heterogeneous networks. In: ECML/PKDD, 2014, pp. 149–164 (2014)
8. Berlingerio, M., Pinelli, F., Calabrese, F.: Abacus: frequent pattern mining-based community discovery in multidimensional networks. Data Min. Knowl. Disc. **27**(3), 294–320 (2013)
9. Chen, Y., Wang, X., Bu, J., Tang, B., Xiang, X.: Network structure exploration in networks with node attributes. Physica A Stat. Mech. Appl. **449**, 240–253 (2016)
10. Chai, B., Yu, J., Jia, C., Yang, T., Jiang, Y.W.: Combining a popularity-productivity stochastic block model with a discriminative content model for general structure detection. Phys. Rev. E **88**, 012807:1–012807:10 (2013)
11. Cheng, H., Zhou, Y., Yu, J.X.: Clustering large attributed graphs: a balance between structural and attribute similarities. ACM Trans. Knowl. Disc. Data **5**, 12:1–12:33 (2011)
12. CoreNLP: Stanford University. https://stanfordnlp.github.io/CoreNLP/. Accessed Oct 2018
13. Camacho, J., Guimerà, R., Amaral, L.: Robust patterns in food web structure. Phys. Rev. Lett. **88**, 228102 (2002)
14. Dan, S., Fusco, J., Shank, P., Chu, K., Schlager, M.: Discovery of community structures in a heterogeneous professional online network. In: System Sciences (HICSS), Hawaii USA, pp. 3262–3271, January 2013
15. Adly, F., et al.: Simplified subspaced regression network for identification of defect patterns in semiconductor wafer maps. IEEE Trans. Ind. Inform. **11**(6), 1267–1276 (2015)
16. Guesmi, S., Trabelsi, C., Latiri, C.: Community detection in multi-relational bibliographic networks. In: Hartmann, S., Ma, H. (eds.) DEXA 2016. LNCS, vol. 9828, pp. 11–18. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44406-2_2
17. Katz, L.: A new status index derived from sociometric analysis. Psychometrika **18**(1), 39–43 (1953)
18. Loe, C.W., Jensen, H.J.: Comparison of communities detection algorithms for multiplex. Phys. A **431**, 29–45 (2015)
19. Taha, K., Yoo, P.: A system for analyzing criminal social networks. In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2015), Paris, France, August 2015
20. Murate, T., Ikeya, T.: A new modularity for detecting one-to-many correspondence of communities in bipartite networks. Adv. Complex Syst. **13**(1), 19–31 (2010)
21. Newman, M.E.J., Clauset, A.: Structure and inference in annotated networks. Nat. Commun. **7**, 11863 (2015)
22. Newman, M.E.J.: Scientific collaboration networks: II. Shortest paths, weighted networks, and centrality. Phys. Rev. E **64**, 016132 (2001)
23. Yoo, P., Muhaidat, S., Taha, K.: Intelligent consensus modeling for proline cis-trans isomerization prediction. IEEE/ACM Trans. Comput. Biol. Bioinform. **11**(1), 26–32 (2014)
24. Protégé: Stanford Center for Biomedical Informatics Research, Stanford University. https://protege.stanford.edu/. Accessed Oct 2018

25. Al-Dalky, R., Taha, K., Al Homouz, D., Qasaimeh, M.: Applying Monte Carlo simulation to biomedical literature to approximate genetic network. IEEE/ACM Trans. Comput. Biol. Bioinform. **13**(3), 494–504 (2016)
26. Taha, K.: GRtoGR: a system for mapping go relations to gene relations. IEEE Trans. Nanobiosci. **12**(4), 1–9 (2013)
27. Sharma, A., Kuang, R., Srivastava, J., Feng, X., Singhal, K.: Predicting small group accretion in social networks: a topology based incremental approach. In: IEEE/ACM International Conference on Advance in Social Networks Analysis and Mining (ASONAM), 2015, pp. 408–415 (2015)
28. Taha, K.: Determining semantically related significant genes. IEEE/ACM Trans. Comput. Biol. Bioinform. **11**(6), 1119–1130 (2014)
29. Taha, K., Elmasri, R.: SPGProfile: speak group profile. Inf. Syst. (IS) **35**(7), 774–779 (2010)
30. Taha, K.: Disjoint community detection in networks based on the relative association of members. IEEE Trans. Comput. Soc. Syst. **5**(2), 493–507 (2018)
31. Tang, J., Zhang, D., Yao, L.: Social network extraction of academic researchers. In: 7th IEEE ICDM, Nebraska, USA, 2007, pp. 292–301 (2007)
32. Taha, K.: Automatic academic advisor. In: 8th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing (IEEE CollaborateCom), Pittsburgh, USA, October 2012
33. Taha, K.: Extracting various classes of data from biological text using the concept of existence dependency. IEEE J. Biomed. Health Inform. (IEEE J-BHI) **19**(6), 1918–1928 (2015)
34. Taha, K., Yoo, P.: SIIMCO: a forensic investigation tool for identifying the influential members of a criminal organization. IEEE Trans. Inf. Forensics Secur. **11**(4), 811–822 (2015)
35. Wang, X., Liu, J.: A layer reduction based community detection algorithm on multiplex networks. Phys. A **471**, 244–252 (2014)
36. Yang, T., Jin, R., Chi, Y., Zhu, S.: Combining link and content for community detection: a discriminative approach. In: 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, France, pp. 927–936 (2009)
37. Yang, J., McAuley, J., Leskovec, J.: Community detection in networks with node attributes. In: IEEE International Conference on Data Mining, 2013, USA, pp. 1151–1156 (2013)
38. Zhu, G., Li, K.: A unified model for community detection of multiplex networks. In: Benatallah, B., Bestavros, A., Manolopoulos, Y., Vakali, A., Zhang, Y. (eds.) Web Information Systems Engineering –WISE 2014, vol. 8786, pp. 31–46. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11749-2_3