Cross-Site Virtual Social Network Construction

Chenhao Xie*, Deqing Yang*, Jingrui He†, Yanghua Xiao*
*School of Computer Science, Shanghai Key Laboratory of Data Science, Fudan University, Shanghai, China
*Email: {xiechenhao,yangdeqing,shawyh}@fudan.edu.cn, Tel: (86)21-51355555
†Arizona State University, AZ, USA. Email: jingrui.he@gmail.com

***Abstract*—**
Given the plethora of social networking sites, it can be difficult for users to browse too many sites and discover social friends. For example, for a new diabetes patient, how can s/he find the users with similar symptoms on different dedicated sites and form supporting groups with them? Since different sites may use different vocabularies, this problem is challenging to match users across different sites. To address it, in this paper, we present a tool to demonstrate how to construct a virtual social network across multiple social networking sites. Specifically, it uses bipartite graphs to represent the relationships between users and their posts' keywords in each site; it bridges the gap between different vocabularies of different sites based on their semantic relatedness through concept-based interpretations; and it uses an efficient propagation algorithm to obtain the similarity between users from different sites, which can be used to construct the cross-site virtual social network.

***Keywords*-cross-site, virtual social network, semantic matching, graph propagation**

I. Introduction

Social networks have experienced fast growth in the last decade, such as Twitter and Facebook. They have become fundamental platforms on which many people maintain their friendships and share information with others. Besides the generic ones, many dedicated social networking sites have been created to help patients with a specific type of disease, such as diabetes. Examples include TuDiabetes (http://www.tudiabetes.org/), Diabetic Connect (http://www.diabeticconnect.com/), Diabetes Sisters (https://diabetessisters.org/), etc. On one hand, these dedicated social networking sites provide diabetes patients rich opportunities to get exposed to recent developments on this disease and to form support groups with people suffering from similar symptoms; on the other hand, once a new user has selected a social networking site, s/he is likely to stick to the site, although it could be the case that many users from another site share a lot of commonalities with this user, thus can provide many useful tips and suggestions.

To help diabetes patients form support groups across multiple websites, we have developed a graph-based system for constructing cross-site virtual social network. It is based on recommendation techniques across heterogeneous domains introduced in [1], of which the goal is to recommend items to users in another heterogeneous domain. In particular, in our system, we build some bipartite graphs to represent the relationships between users of each site and the keywords used in their posts. Then, to bridge the gap between different vocabularies used by different sites, we infer their semantic relatedness through concept-based interpretation distilled from online encyclopedias, such as Wikipedia. Finally, the similarities between users of two different sites are computed as similarity scores via an efficient graph propagation algorithm. Such similarity scores can be used to construct a cross-site virtual social network for the sake of forming support groups for the diabetes patients.

Our techniques are different from: (1) existing work on cross-domain recommendation [2] in the sense that we target heterogeneous domains with barely overlapping feature sets (vocabularies); and (2) transfer learning [3], [4] across heterogeneous domains as we aim to build the connections between users across different sites instead of learning multiple predictive models.

To thoroughly demonstrate our techniques on constructing cross-site virtual social network, we have implemented a graphic tool which will be presented in the following section.

The rest of this paper is organized as follows. In Section II, we introduce the key techniques used in our system, followed by introducing the user interface and functionality of our demonstration tool in Section III. And then we present our paper's related work in Section IV. Finally, we conclude the paper in Section V.

II. Key Techniques

In this section, we introduce our graph-based recommendation system based on [1]: we start with some notations, followed by the introduction of the global similarity and the efficient computation of the relevance vectors, and finally discuss semantic matching for bridging the gap between different vocabularies used by different social networking sites.

A. Notation

In this paper, for the sake of clarity, we consider 2 different sites. Formally, for the i^{th} domain (i=1,2), we use a bipartite graph $G_i = \{V_i, E_i\}$ to represent the relationships between the users of one site and the keywords used in users' historical posts, where V_i is the set of nodes in this graph, and E_i is a set of undirected edges. Let n_i denote the number of users in the i^{th} site, and m_i denote the number of keywords. Therefore, V_i consists of two types of nodes: n_i user nodes, and m_i keyword nodes. Let $\boldsymbol{X}_i$, $n_i \times m_i$, denote

This demo was supported by NSFC (No.61472085, 61171132, 61033010), by National Key Basic Research Program of China under No.2015CB358800, by Basic research project of Shanghai science and technology innovation action plan under No.15JC1400900, and by Shanghai Science and Technology Development Funds (13dz2260200, 13511504300). Corresponding author is Yanghua Xiao.

the connectivity matrix between the two types of nodes, whose elements are set to be the edge weights (e.g., the TF-IDF value of the keywords to a user). Furthermore, let $G_0 = \{V_0, E_0\}$ denote the bipartite matching graph, where V_0 includes all the keyword nodes from the two sites, and E_0 denotes the set of edges connecting keywords from different sites. Based on this graph, we define a connectivity matrix $\boldsymbol{X}_0$, $(m_1+m_2)\times(m_1+m_2)$, whose elements measure the similarity between features from different domains. Details of learning the connectivity matrix $\boldsymbol{X}_0$ based on semantic matching will be discussed in Subsec. II-C.

Putting all above graphs together, we get a multi-partite graph $G = \{V, E\}$ as shown in Figure 1, where $V = V_1 \bigcup V_2$, and $E = E_1 \bigcup E_2 \bigcup E_0$. Then, we define an *affinity matrix* $\boldsymbol{X}$ for this graph. $\boldsymbol{X}$ is an $(n_1+n_2+m_1+m_2)\times(n_1+n_2+m_1+m_2)$ matrix and is represented as,

$$\boldsymbol{X} = \begin{bmatrix} \mathbf{0}_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \boldsymbol{X}_1 & \mathbf{0}_{n_1\times m_2} \\ \mathbf{0}_{n_2\times n_1} & \mathbf{0}_{n_2\times n_2} & \mathbf{0}_{n_2\times m_1} & \boldsymbol{X}_2 \\ \boldsymbol{X}_1^T & \mathbf{0}_{m_1\times n_2} & \mathbf{0}_{m_1\times m_1} & \boldsymbol{X}_0 \\ \mathbf{0}_{m_2\times n_1} & \boldsymbol{X}_2^T & \boldsymbol{X}_0^T & \mathbf{0}_{m_2\times m_2} \end{bmatrix}$$

where $(\cdot)^T$ denotes matrix transpose and $\mathbf{0}$ is a zero matrix. Based on this graph, our goal is to infer the similarities between user nodes from different domains.

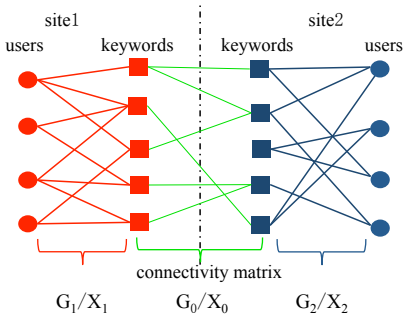

Figure 1. A multi-partite graph across two sites. Red lines and blue lines are user-keyword edges in site1 and site2, respectively. The green lines between red rectangles and blue rectangles are the edges between keywords from different sites, which can be established by the semantic similarities of keywords.

B. Global Similarity between Objects

In graph G, the direct links between user nodes from different sites are absent. Thus, to establish such links, we will measure the similarities between these users based on a graph propagation algorithm. To be specific, we first normalize the affinity matrix $\boldsymbol{X}$ as follows.

$$\boldsymbol{S} = \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{X}\boldsymbol{D}^{-\frac{1}{2}} = \begin{bmatrix} \mathbf{0}_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} & \boldsymbol{S}_1 & \mathbf{0}_{n_1\times m_2} \\ \mathbf{0}_{n_2\times n_1} & \mathbf{0}_{n_2\times n_2} & \mathbf{0}_{n_2\times m_1} & \boldsymbol{S}_2 \\ \boldsymbol{S}_1^T & \mathbf{0}_{m_1\times n_2} & \mathbf{0}_{m_1\times m_1} & \boldsymbol{S}_0 \\ \mathbf{0}_{m_2\times n_1} & \boldsymbol{S}_2^T & \boldsymbol{S}_0^T & \mathbf{0}_{m_2\times m_2} \end{bmatrix} \quad (1)$$

where $\boldsymbol{D}$ is a diagonal matrix with each element equal to the row sum of $\boldsymbol{X}$; $\boldsymbol{S}_1$, $\boldsymbol{S}_2$, and $\boldsymbol{S}_0$ are normalized versions of $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, and $\boldsymbol{X}_0$, respectively.

In order to compute the global similarities between the i^{th} node and all the other nodes in the composite multi-partite graph, we use $\boldsymbol{v_i}$ to denote an $n_1+n_2+m_1+m_2$ dimensional vector, whose i^{th} element is 1 and all the others are 0. Thus the global similarity vector with respect to the i^{th} node can be written as $(\boldsymbol{I}-\alpha\boldsymbol{S})^{-1}\boldsymbol{v}_i$, where $\boldsymbol{I}$ is an $(n_1+n_2+m_1+m_2)\times(n_1+n_2+m_1+m_2)$ identity matrix, and α is a positive scalar whose value is close to 1. Putting all these vectors together, we have the following global similarity matrix $\boldsymbol{K}$ which has $(n_1+n_2+m_1+m_2)\times(n_1+n_2+m_1+m_2)$ dimensions,

$$\boldsymbol{K} = (\boldsymbol{I}-\alpha\boldsymbol{S})^{-1}[\boldsymbol{v}_1,\ldots,\boldsymbol{v}_{n_1+n_2+m_1+m_2}] = (\boldsymbol{I}-\alpha\boldsymbol{S})^{-1} \quad (2)$$

It is easy to see that $\boldsymbol{K}$ has the following block structure,

$$\boldsymbol{K} = \left(\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{C} \end{bmatrix}\right)^{-1} = \begin{bmatrix} \boldsymbol{K}_1 & \boldsymbol{K}_2 \\ \boldsymbol{K}_2^T & \boldsymbol{K}_3 \end{bmatrix}$$

where $\boldsymbol{K}_1$, $\boldsymbol{K}_2$, and $\boldsymbol{K}_3$ are submatrices of $\boldsymbol{K}$,

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{I}-\mathbf{0}_{n_1\times n_1} & \mathbf{0}_{n_1\times n_2} \\ \mathbf{0}_{n_2\times n_1} & \boldsymbol{I}-\mathbf{0}_{n_2\times n_2} \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} -\alpha\boldsymbol{S}_1 & \mathbf{0}_{n_1\times m_2} \\ \mathbf{0}_{n_2\times m_1} & -\alpha\boldsymbol{S}_2 \end{bmatrix}$$

and

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{I} & -\alpha\boldsymbol{S}_0 \\ -\alpha\boldsymbol{S}_0^T & \boldsymbol{I} \end{bmatrix}.$$

Since we are only interested in the similarities among users from different sites, instead of the whole matrix $\boldsymbol{K}$, we only compute the submatrix $\boldsymbol{K}_1$, $(n_1+n_2)\times(n_1+n_2)$, which has the following closed form solution.

$$\boldsymbol{K}_1 = \left(\boldsymbol{I}-\boldsymbol{B}\boldsymbol{C}^{-1}\boldsymbol{B}^T\right)^{-1} \quad (3)$$

Based on $\boldsymbol{K}_1$, the relevance between the k^{th} user and all the users from the other site can be obtained from the k^{th} row or column of $\boldsymbol{K}_1$, since $\boldsymbol{K}_1$ is a symmetric matrix. In other words, the *relevance vector* can be written as $\boldsymbol{s}_i = \boldsymbol{K}_1\boldsymbol{u}_i$, where $\boldsymbol{u}_i$ is an n_1+n_2 dimensional vector. The elements of $\boldsymbol{u}_i$ are 0 except the i^{th} one, which is set to 1. An efficient algorithm for computing the relevance vector was introduced in [1], which is based on an iterative power method.

C. Semantic Matching

According to our algorithm, some relationships between keywords should be established in order to discover the similarity of users from the two social networking sites, i.e., inferring E_0 in G_0. In this subsection, we introduce how to find the relationships between different keywords from two sites, respectively.

As we known, an online encyclopedia contains millions of concepts including person, location, organization, hobby and etc. There exists an article page describing the fact about each concept. As well, there are many hyper-links linking to other concepts on each article page to enrich its semantic description. According to the principle of previous Wikipedia based methods [5], [6], two terms, i.e., two concepts, are considered semantically related if they co-occur as hyper-links in one article page. The more such article pages can be found, the more semantically related

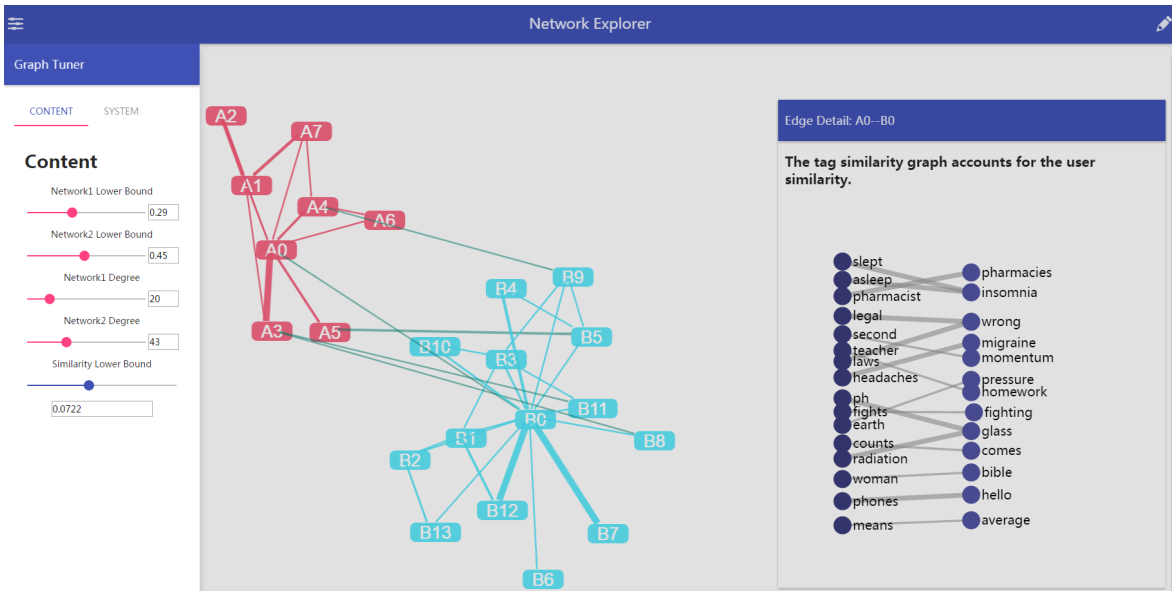

Figure 2. Screenshot of demonstration tool's user interface. Center main frame displays two social networks including cross-site social links issued by our algorithm. Left content panel is used to control the display of nodes and edges in main frame. Right panel shows semantic relatedness between keywords of two users.

the two terms are. In our scenario, a keyword equals to a concept if they are represented by the same term.

According to ESA's (Explicit Semantic Analysis) basic idea [5], the semantic interpretation of a keyword i can be represented by a *concept vector* which is formally defined as $\vec{C_i} = [c_1, ..., c_C] \in \mathbb{R}^C$. C is the total number of concepts in the encyclopedia and $c_j (1 \leq j \leq C)$ represents the semantic relevance of concept j to keyword i, i.e., the TF-IDF score of concept i (represented by keyword i) in concept j's article. Thus, the semantic similarity between keyword a and keyword b can be represented as the cosine similarity of $\vec{C_a}$ and $\vec{C_b}$, namely $cos(\vec{C_a}, \vec{C_b})$. To establish the links between keywords from the two sites, i.e., E_0 in the bipartite graph G_0, we can set a threshold λ. Then, the edge between a and b is created if $cos(\vec{C_a}, \vec{C_b}) \geq \lambda$. As well, $cos(\vec{C_a}, \vec{C_b})$ is set as the element value of connectivity matrix $\boldsymbol{X}_0$ (refer to Figure 1). Accordingly, λ decides the density of $\boldsymbol{X}_0$. At last, we can compute $\boldsymbol{s}_i$ through the algorithm introduced in Subsec. II-B when $\boldsymbol{X}_0$ is inferred.

III. User Interface and Functionality

In this section, we demonstrate the user interface and functionality of our graphic tool which can be accessed through http://218.193.131.244:8000. The tool was coded by Java and can be viewed in Google Chrome and Firefox explorer. Currently, our tool is illustrated by using the dataset of two diabetes social network websites, i.e., http://www.tudiabetes.org (denoted by Diabetes1) and https://diabetessisters.org (denoted by Diabetes2). The former site is dedicated to diabetes patients of Type I, Type II, and prediabetes, and the latter focuses on female diabetes patients, especially those with gestational diabetes.

Our goal is to establish virtual social links between the users from Diabetes1 and Diabetes2, respectively. Our tool was designed not only to display the virtual social links between the users from Diabetes2 and Diabetes1 respectively, but also to reflect the similarities between the users in one website. In each diabetes website, we use the similarity of two users' keyword sets to represent the similarity between these two users. Specifically, one user is represented by a keyword vector in which each element is the TF-IDF value of one keyword to him/her, i.e, the edge weight in $G_i (i = 1, 2)$. Then, we compute the cosine similarity of the two vectors as the two users' similarity. Such similarity can be used as the ground to issue the social links between the users in one site. The similarity between two users from the two diabetes websites respectively is computed according to the algorithm in Subsec. II-B.

Our tool's user interface is shown in Figure 2. The main frame is in center of the interface where the users from the two websites are displayed as red nodes and blue nodes, respectively. The red edges and blue edges are the social relationships built based on user similarities in each website. The green edges between the nodes from the two websites respectively are the virtual social relationships established by our algorithm. For all the three types of edges, each edge's thickness represents its weights, i.e., the similarity between the two ends (users) of the edge.

In the left content panel, all scroll bars are used to control different thresholds that adjust the display of nodes and edges in the main frame. At first, in each website, an edge can be accounted and in turn, be displayed in the main frame only when its weight is bigger than 'Network1/2 Lower Bound'. 'Similarity Lower Bound' has the same function to control the display of green edges. For a node in each website, its degree is the number of the accounted edges linking it to other nodes in the same site. Then, one node is displayed in the main frame when its degree is bigger than 'Network1/2 Degree'.

The right panel displays the details about the semantic relatedness between two keyword sets, which are used to infer the similarity between two users from the two websites, respectively. For example, when we click the green edge between A0 and B0, the right panel will show A0's keywords as left column and B0's keywords as right column. All keywords are positioned according to their similarities (TF-IDF value) to the user. And the grey lines between the two columns demonstrate the semantic relatedness between the keywords belonging to A0 and B0 respectively, which are computed based the algorithm in Subsec. II-C. Moreover, the thickness of each line can indicate semantic relatedness value between the two keywords.

IV. Related Work

In this section, we survey the research works related to the techniques built in our demonstration tool.

A. Semantic Relatedness Measurement

Many prior works on measuring semantic relatedness of two terms utilized the lexical concepts in WordNet's taxonomy [13] based on the deepest point in the taxonomy [14] or information content [15]. To expand concept coverage, many researchers took Wikipedia as the knowledge base of semantic interpretation. M. Strube et al. [16] and D.Milne et al. [6] used the taxonomy and the Normalized Google Distance [17] in Wikipedia to compute semantic relatedness, respectively. E.Gabrilovich et al. [5] proposed a widely applied model of semantic interpretation, i.e., Explicit Semantic Analysis which is also based on the relations between concepts in online encyclopedia.

B. Cross-domain Recommendation

In fact, constructing virtual social links across different websites can be viewed as cross-domain friend recommending. Ignacio et al. [7] proposed a survey of emerged solutions for cross-domain recommendation and emphasized two major tasks. Many previous works on cross-domain recommendation focus on improving CF-based scheme. For example, the authors in [2], [8] tried to migrate the rating data from a dense auxiliary domain to alleviate the cold start problem in a sparse target domain resulting in the improvement of recommendation performance in the target domain. Besides, [9], [10] merged user profiles distributed in different domains for better recommendation.

C. Transfer Learning

Transfer learning aims to improve a learning task in a target domain by using the knowledge transferred from other domain in which a related task is known [11]. Recently, transfer learning techniques have been widely applied to mitigate the sparsity problem of collaborative filtering in cross-domain recommendation. In [2], Li et al. proposed a transfer learning approach that performs a co-clustering strategy on the rating matrix of an auxiliary domain with high rating density, and discovers rating patterns at the cluster level. Y. Zhu et al. [12] applied transfer learning method to learn image classification by using the knowledge of document/image labels in auxiliary domains. In this work, relations between documents and images are captured by their co-occur tags. These methods are not suitable to be applied in our setting, where we aim to build the connections between heterogeneous users (with different keywords) across different websites instead of learning multiple models.

V. Conclusions

In this paper, we demonstrate a graphic tool which was designed to construct virtual social relationships across different social networking sites. To this end, we not only propose an algorithm to discover the semantic relatedness between different vocabularies, but also design an efficient propagation algorithm to capture the similarities between users from different sites. Our tool can clearly display the connections between different entities (users and keywords) among the networks.

References

[1] D. Yang, J. He, H. Qin, Y. Xiao, and W. Wang, "A graph-based recommendation across heterogeneous domains," in *Proc. of CIKM*, 2015.

[2] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction," in *Proc. of IJCAI*, 2009.

[3] Z. Lu, E. Zhong, L. Zhao, E. Xiang, W. Pan, and Q. Yang, "Selective transfer learning for cross domain recommendation," in *Proc. of SDM*, 2013.

[4] B. Cao, N. N. Liu, and Q. Yang, "Transfer learning for collective link prediction in multiple heterogenous domains," in *Proc. of ICML*, 2010.

[5] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proc. of IJCAI*, 2007.

[6] D. Milne and I. H. Witten, "An effective, low-cost measure of semantic relatedness obtained from wikipedia links," in *Proc. of AAAI*, 2008.

[7] I. Fernandez-Tobias, I. Cantador, M. Kaminskas, and F. Ricci, "Cross-domain recommender systems: A survey of the state of the art," 2011.

[8] W. C. Wynne, H. Mong, and L. Lee, "Making recommendations from multiple domains," in *Proc. of SIGKDD*, 2013.

[9] M. Szomszor, H. Alani, I. Cantador, K. OHara, and N. Shadbolt, "Semantic modelling of user interests based on cross-folksonomy analysis," in *Proc. of ISWC*, 2008.

[10] F. Abel, E. Herder, G.-J. Houben, N. Henze, and D. Krause, "Cross-system user modeling and personalization on the social web," in *Proc. of UMUAI*, 2013.

[11] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE TKDE*, vol. 22, no. 10, pp. 1345 - 1359, 2010.

[12] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G.-R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification," in *Proc. of AAAI*, 2011.

[13] F. L., G. E., and Matias, *WordNet: An Electronic Lexical Database*. Cambridge, MA.: MIT Press, 1998.

[14] Z. Wu and M. Palmer, "Verb semantics and lexical selection," in *Proc. of ACL*, 1994.

[15] P. Resnick, "Using information content to evaluate semantic similarity in a taxonomy," in *Proc. of IJCAI*, 1995.

[16] M. Strube and S. P. Ponzetto, "Wikirelate! computing semantic relatedness using wikipedia," in *Proc. of AAAI*, 2006.

[17] R. L. Cilibrasi and P. M. Vitanyi, "The google similarity distance," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 3, pp. 370–383, 2007.