Multi-Relational Link Prediction for an Online Health Community

Sulyun Lee[*], Hankyu Jang[†], Kang Zhao[‡], Michael S. Amato[§], and Amanda L. Graham[§]

[*] *Interdisciplinary Graduate Program in Informatics, University of Iowa*
[†] *Department of Computer Science, University of Iowa*
[‡] *Department of Business Analytics, University of Iowa*
[§] *Innovations Center, Truth Initiative*
[*, †, ‡] {sulyun-lee, hankyu-jang, kang-zhao}@uiowa.edu
[§] {mamato, agraham}@truthinitiative.org

Abstract

Social networks often incorporate multiple types of social relationships, making them multi-relational networks. Effective link predictions can help social networks improve user experience and engagement, but limited attention has been paid to predicting links in multi-relational networks. This paper explores link predictions in multi-relational networks from an online health community. We demonstrate that leveraging information from multiple networks built based on different types of relationships is superior to using only information from a single network or the aggregated network. In addition, adding community structures, nodal similarities based on network embedding and topic similarity can help link predictions in different ways. Our work has implications for the design and management of a successful online health community.

Keywords— Social Network, Link Prediction, Community Detection, Network Embedding, Multi-Relational Network, Supervised Learning, Online Health Community

1 Introduction

Online health communities (OHCs) enable social networking among those with similar health concerns to get social support and helpful information [11]. Such online social networks often offer various types of communication channels to facilitate user interactions. For example, a messaging channel allows individuals to send direct messages in a one-to-one fashion, whereas in a discussion channel, users post content and reply to others' posts in public communications.

Meanwhile, such online social networks grow over time as more connections are created among the users. Based on the information from the current network, link predictions can detect the features of observed network ties and infer to where future ties would form [7]. Effective predictions of future social network connections can help an OHC build recommender systems. Such systems can help users find those that deem interesting and thus keep them engaged in the community, which is one of the keys for a successful OHC [17].

In our study, we use data from a popular OHC for smoking cessation, BecomeAnEx. This OHC provides four types of communication channels–blogs&comments, group discussions, message boards, and private messages. For each of these channels, we construct one network based on users' interactions through that channel. Thus the four networks, one for each channel, constitute a multi-relational network, where the same set of nodes are connected by edges that represent different types of relationships [16, 18]. Potential interconnections among different types of relationships in a multi-relational network may have made it more complicated to predict links in such a network. At the same time, a multi-relational network also offers opportunities to leverage more fine-grained information from different types of relationships for link predictions, because users' interactions via one type of relationship may affect link formations based on another type. Nevertheless, the majority of existing link prediction methods do not take advantages of these dependencies into consideration [8].

Specifically, we propose a supervised link prediction approach for multi-relational networks in an OHC. We investigate the value of utilizing different types of edges in a multi-relational network. Moreover, our approach also incorporates community structures, nodal similarities based on network embeddings, as well as textual similarities.

2 Related Work

Davis et al. (2011) [6] proposed a link prediction method for heterogeneous networks, which have different types of nodes and edges. Their approach is based on enumerate patterns of the existence and absence of different types of edges in triads. However, due to its computational complexity, the approach can hardly be adopted for multi-relational networks with more than 3 types of relationships.

Inspired by the idea that homogeneous networks may contain multi-relational information, Wang et al. proposed an approach of constructing a weighted network that contains heterogeneity information [15]. They constructed a multi-relational network applying Edge Clustering[14], which clusters edges to maximize the modularity of the network. The resulting top eigenvectors of the modularity matrix were adopted as the social features for each node. Cosine similarities between the social features for node pairs are computed which are assigned as edge weights and this weighted network is used in supervised link prediction. Though their approach is generally applicable to any network, multi-relational information cannot be adopted into this approach.

3 Data and Setup

3.1 Data Source

The dataset for this paper comes from BecomeAnEx, one of the most popular OHCs for smoking cessation. The data includes users' online interactions for 6 years, from 2010 to 2015. Users' interactions occurred in four communication channels: blogs&comments (BC), group discussions (GD), message boards (MB), and private messages (PM). In our networks, nodes denote users of the OHC and edges indicate the interactions between users. Our multi-relational network consists of four networks among the same set of nodes, one for each communication channel. In BC and GD, one user publishes a blog post or start a discussion thread, then other users then post comments to the blog post or thread that they are interested in participating. For these two channels, we created a link between the user who published the blog post or started the discussion thread and those

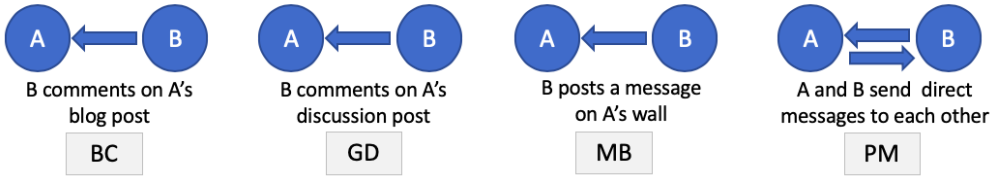

Figure 1: Four communication channels in BecomeAnEx

who replied. In the MB channel, one user can post a public message on another user's message board, which is similar to a Facebook wall. In this case, posting such a message creates a link between the owner of the wall and the user who posted to the wall. For the PM channel, which represents one-to-one communication, if a user sends a private message to another user and receives a message back from the user, a link is formed. Since administrators and senior members of OHC often send welcome or greeting messages, which are often not reciprocated, we establish a link between two users only if the communications between two users are reciprocal. Figure 1 depicts how we constructed networks from our data.

Though our networks have directed links among users, we considered it as an undirected network and for our link prediction problem as undirected because when it comes to social support, both seeker and providers can benefit from such activities. Also, we did not consider the weight of the edges.

We selected 52 consecutive weeks of user interactions, from week 50 to week 101, when users were most active during the time span of our dataset. For instance, four initial networks, G_{BC50}, G_{GD50}, G_{MB50}, and G_{PM50} are constructed based on users' activities during week 50. In addition, we also constructed an aggregated network G_{AGG50} that aggregates all of the user interactions across the four channels–as long as two nodes are connected in one of the four networks, they are connected in the aggregated network. Table 1 shows the statistics of the five networks mentioned above.

Table 1: Network statistics of each channel

Network property	G_{BC}	G_{GD}	G_{MB}	G_{PM}	G_{AGG}
Number of nodes	16114	2246	3988	497	20096
Number of edges	98518	3254	13079	971	115084
Average degree	12.228	2.898	6.559	3.907	11.453
Maximum degree	8334	162	778	100	9032
Degree standard deviation	100.946	6.041	31.648	8.866	103.079
Clustering coefficient	0	0	0.203	0.149	0.518
Number of connected components	16	47	24	39	51
Assortativity	-0.212	-0.140	-0.358	-0.208	-0.218

[a]The graphs are constructed with the user interactions that occurred anytime during the weeks from 50 to 101.

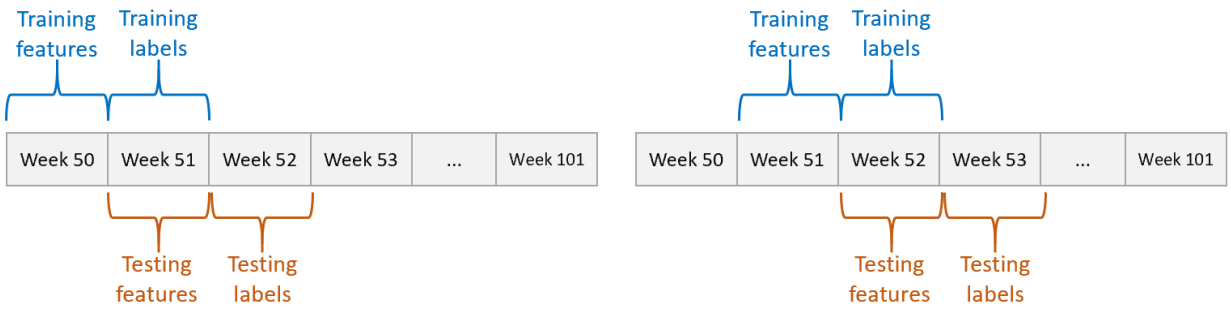

Figure 2: Description of training and testing set construction

3.2 Setup

We set up the link prediction on a weekly basis–predicting if two currently disconnected nodes will form a new tie during the next week based on what is observed during the current week. To make the dataset more balanced, we only included node pairs (i.e., instances) that are two hops away in the aggregated network during training week t, $G_{AGG\cdot t}$. Labels of a training instance were set to 1 if the two corresponding nodes form a new tie during the week $t+1$ in $G_{AGG\cdot t+1}$, and 0 otherwise. Features for the training set were extracted based on networks at week t. Networks for generating features vary with respect to which set of features we are using, to be discussed later in section 4.

A similar approach was used to generate the testing set of week t. Instances are two-hop node pairs in the aggregated network constructed for a week $t+1$, $G_{AGG\cdot t+1}$, excluding those that are already connected in the aggregated network of the week t, $G_{AGG\cdot t}$. Testing set features were based on the snapshot of the networks in week $t+1$, again networks used to generate features vary across different feature set we use. The testing set labels were based on tie formation during the week $t+2$ in $G_{AGG\cdot t+2}$. In summary, we are using the user interaction in one or more channels in the OHC, and utilizing the information to predict net pair of users that would interact in any of the channels in OHC in the following week. Figure 2 describes the data construction procedure.

4 Method

4.1 Baselines

The 3 baseline features that we used in this experiment all attempt to capture proximity between nodes and have been widely adopted in the link prediction literature. Jaccard coefficient (JC) captures the number of common neighbors of two nodes divided by the number of total neighbors of the two nodes. Preferential attachment (PA) uses the degree multiplication of two nodes as the similarity value of two nodes. Adamic Adar (AA) holds the summation of the inverse-log degree of common neighbors of two by assigning more weights to the neighbors with a lower degree.

4.2 Multi-Relational Link Prediction

The baseline model for our experiments does not consider a social network as multi-relational and thus extracts the 3 baseline features for the aggregated network only, yielding feature set F_{AGG}. We propose a multi-relational link prediction (MRLP) approach. It considers nodal proximity across the four networks and extracts the 3 baseline features for each of the communication channels, yielding feature set F_{ALL} that consists of F_{BC}, F_{GD}, F_{MB}, F_{PM} generated from G_{BC}, G_{GD}, G_{MB},

G_{PM}, and G_{AGG} respectively. This adjustment in generating features for a multi-relational network makes it feasible for classifiers to learn characteristics of nodal proximity from each channel and leverage information from different types of edges.

4.3 Adding More Features

In addition to the 3 baseline features, we also introduce additional features, namely community-based features, embedding-similarity features, and text-similarity features, and evaluate if they improve the link prediction performance for the OHC.

4.3.1 Community-Based Feature

Community-based features capture if two nodes belong to the same network community. This is based on the assumption that nodes in the same network community have a higher chance of interacting with each other. Among several community detection algorithms, we selected two computationally efficient algorithms where the number of communities is determined by the algorithm. For each algorithm, we generated a binary feature for each pair of nodes to indicate if the two are in the same network community. We denoted the feature set that includes community-based binary features generated on each communication channel as F_{COM}.

The community detection algorithm based on modularity maximization (CM) is proposed by Clauset et al. [3]. We used C_i, $1 \leq i \leq k_{CM}$, to denote k_{CM} number of communities detected by this algorithm. If nodes x and y are in the same community C_i, the similarity between x and y is set to 1, and 0 otherwise.

$$s_{xy} = \begin{cases} 1, & \text{if } x, y \in C_i,\ (1 \leq i \leq k_{CM}) \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

The other community detection algorithm we used is based on label propagation (CLP) which is proposed by Cordasco and Gargano [4]. In this algorithm, each node is initialized with a unique community, and in each iteration, each node is combined with the nodes with the majority of neighborhood communities. The membership of a node is updated to that of its neighboring node randomly if no community dominates the others. The number of communities, k_{CLP}, is also determined by the algorithm. The notion of similarity is the same as the CM mentioned above:

$$s_{xy} = \begin{cases} 1, & \text{if } x, y \in C_i,\ (1 \leq i \leq k_{CLP}) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

4.3.2 Embedding-Similarity Feature

Network embedding learns vector representations of nodes in the networks [5]. The learned representations can reflect the structural and neighborhood properties of each node of the network. With such vector representations, similarity can be calculated between each pair of nodes. Among several ways of generating network embedding, we applied DeepWalk (Perozzi et al., 2014) [13].

DeepWalk incorporates the language modeling scheme to the network, considering each node as a word in a corpus. In the language modeling using skip-gram model, it maximizes the probability of having the next word in a corpus, given the sequence of previous words. Likewise, in DeepWalk, it first performs a series of random walks from a source node to produce a sequence of nodes. Then, it maximizes the probability of predicting the next node, given the previous nodes generated

by the random walks. Therefore, the learned vector representations of the nodes can obtain the neighborhood information by randomly searching for adjacent nodes.

For the parameters used in DeepWalk algorithm, we followed the experiment settings described in the work of Perozzi et al. [13] across all of our experiments to ensure the fair comparison and decent performances. Given a graph, we generated a sequence of random walks having the length of 40 walks for each source node, and we picked 128 to be the size of embedding vectors. We repeated the walks 80 times at the starting of each node. Using a sliding context size of 21, we applied the skip-gram model that maximizes the likelihood of predicting the next coming nodes, given the previous node in the context. Once we obtained the final vector representations with size 128 for all the nodes in the graph using DeepWalk algorithm, we then computed cosine similarities for every node pair. Specifically, given a node pair x and y, we learned the vector representations V_x and V_y for the nodes and computed the cosine similarity. The feature set based on the embedding-similarity features for each channel is denoted as F_{EMB}.

$$s_{xy} = cos(V_x, V_y) \tag{3}$$

4.3.3 Text-Similarity Feature

Users who care about similar topics in an OHC may have a higher chance of interacting with each other. Thus we calculated textual similarity among users' posts as a measure of nodal similarity. Specifically, we combined all the posts during the 50 week period across different channels except for PM since the contents of private messages were excluded from this study for privacy concerns. Texts were pre-processed by removing the stop words, lemmatizing, and stemming. Then, we applied the latent Dirichlet allocation (LDA) [1] with 30 topics to model the topic distribution of each post. Note that our focus is topical similarity, which is usually robust against the choice of the number of topics.

For post n published by user i in channel j, during week t, its topic distribution is denoted as T_{ijt}. Since our prediction is on a weekly basis, to represent the topic distribution of each user in a specific channel, we averaged the topic distributions of the user's posts during the corresponding week and via the channel.

Then, we computed the cosine similarity between the averaged topic distributions for a pair of the user to get text-based nodal similarity. In this way, if the averaged topic distributions of the posts are similar in the same week and the same channel, we regarded that the two users are interested in a similar topic, yielding a high similarity. The text-similarity for nodes x and y can be computed as follows:

$$s_{xy} = cos(A_x, A_y), \text{ where } A_i = \frac{\sum_{t=1}^{n} T_{ijt}}{n} \tag{4}$$

We used feature set F_{TEX} to represent text-similarity features based on each channel.

5 Results

We evaluated four different classifiers to compare the performance of the different models and different feature sets. Note that we did not tune parameters of classifiers for each set of features since our goal is to compare the predictive power of different models and feature sets rather than to optimizing link predictions. We used the default parameter settings from [12] for the random forest, logistic regression, and AdaBoost in classification. Neural network was trained with one

hidden layer with 32 neurons on the dataset using mini-batch gradient descent with a batch size of 20 for ten epochs by setting aside 20 percent of the data per epoch [2].

Classification results were then evaluated with precision and precision@K as our goal is to recommend top future links with high accuracies instead of recovering all future links. We also included normalized discounted cumulative gain (nDCG) as an evaluation measure as it is also important to rank links that occurred higher than those that did not occur.

Link prediction results on baseline model and MRLP, both with 3 baseline features, is summarized in Table 2. Each value in the table is the average of the prediction results across 50 weeks of the dataset, where the values in bold denote the best performer for each evaluation metric. It is clear that MRLP performs better than other baseline approaches: its precision, precision@10, and precision@20 are 25%, 2%, and 20% better than the best performing baseline model, respectively. Its nDCG@10 and nDCG@20 are 10%, and 15% better than the best performing baseline model respectively. In other words, considering each network works better than considering a single network or aggregating these networks into one.

Table 3 illustrates the value of additional features in our multi-relational link prediction approach. The values in the table are the averaged performance across all 50 weeks, as in the previous table. Across all evaluation metrics, MRLP + F_{EMB} performs the best among the additional features, outperforming MRLP. The precision, precision@10, and precision@20 are 34%, 26%, and 11% higher in MRLP + F_{EMB} than MRLP respectively. The nDCG@10 and nDCG@20 for MRLP

Table 2: Results for Baseline vs. MRLP

Metric	**Classifier**	**Baseline Approach**					**MRLP**
		F_{AGG}	F_{BC}	F_{GD}	F_{MB}	F_{PM}	F_{ALL}
Precision	Random Forest	0.019	0.001	0.001	0.063	0.032	0.055
	Logistic Regression	0.071	0.000	0.029	0.093	0.031	**0.116**
	AdaBoost	0.080	0.000	0.001	0.048	0.005	0.101
	Neural Network	0.007	0.000	0.000	0.000	0.011	0.030
PREC@10	Random Forest	0.022	0.004	0.006	0.056	0.030	0.040
	Logistic Regression	0.136	0.000	0.051	0.168	0.064	0.168
	AdaBoost	0.102	0.004	0.006	0.144	0.026	**0.184**
	Neural Network	0.128	0.000	0.012	0.180	0.058	0.172
nDCG@10	Random Forest	0.020	0.003	0.009	0.054	0.033	0.039
	Logistic Regression	0.127	0.000	0.045	0.170	0.079	0.156
	AdaBoost	0.110	0.003	0.007	0.155	0.026	**0.197**
	Neural Network	0.126	0.000	0.013	0.179	0.066	0.178
PREC@20	Random Forest	0.019	0.002	0.012	0.060	0.025	0.046
	Logistic Regression	0.102	0.002	0.047	0.144	0.068	0.160
	AdaBoost	0.088	0.004	0.011	0.130	0.020	0.169
	Neural Network	0.106	0.000	0.028	0.160	0.057	**0.192**
nDCG@20	Random Forest	0.019	0.002	0.012	0.058	0.029	0.043
	Logistic Regression	0.106	0.001	0.044	0.153	0.077	0.155
	AdaBoost	0.097	0.003	0.010	0.142	0.022	0.181
	Neural Network	0.112	0.000	0.024	0.166	0.063	**0.191**

[a]Values that are in bold denote the largest value for each evaluation metric.

$+ F_{EMB}$ are 23% and 18% better than MRLP. These results indicate that incorporating the embedding features in addition to the base features in MRLP improved the prediction power of the classifiers. MRLP + F_{COM} also achieved increased performance compared to MRLP across all evaluation metrics.

Table 3: Performance of additional features on MRLP

Metric	**Classifier**	**MRLP**	**MRLP+More Feature Sets**				
		F_{ALL}	F_{ALL} $+F_{COM}$	F_{ALL} $+F_{EMB}$	F_{ALL} $+F_{TEX}$	F_{ALL} $+F_{COM}$ $+F_{EMB}$	F_{ALL} $+F_{COM}$ $+F_{EMB}$ $+F_{TEX}$
Precision	Random Forest	0.055	0.035	0.050	0.030	0.043	0.056
	Logistic Regression	0.116	0.101	0.136	0.102	0.123	0.122
	AdaBoost	0.101	0.148	**0.155**	0.122	0.143	0.136
	Neural Network	0.030	0.029	0.050	0.013	0.043	0.085
PREC@10	Random Forest	0.040	0.040	0.048	0.048	0.040	0.044
	Logistic Regression	0.168	0.192	0.140	0.168	0.164	0.188
	AdaBoost	0.184	0.194	0.162	0.144	0.136	0.138
	Neural Network	0.172	0.160	**0.232**	0.164	0.184	0.168
nDCG@10	Random Forest	0.039	0.040	0.052	0.051	0.041	0.045
	Logistic Regression	0.156	0.189	0.139	0.158	0.157	0.183
	AdaBoost	0.197	0.206	0.177	0.143	0.142	0.143
	Neural Network	0.178	0.165	**0.243**	0.166	0.184	0.172
PREC@20	Random Forest	0.046	0.041	0.049	0.053	0.039	0.047
	Logistic Regression	0.160	0.198	0.166	0.148	**0.194**	0.156
	AdaBoost	0.169	0.152	0.150	0.137	0.144	0.144
	Neural Network	0.192	0.158	**0.214**	0.186	0.170	0.150
nDCG@20	Random Forest	0.043	0.041	0.051	0.054	0.040	0.047
	Logistic Regression	0.155	0.195	0.158	0.148	0.181	0.162
	AdaBoost	0.181	0.171	0.163	0.139	0.147	0.146
	Neural Network	0.191	0.163	**0.226**	0.180	0.175	0.158

[a]Values that are in bold denote the largest value for each evaluation metric.

6 Conclusions and Future Work

In this paper, we proposed an approach for supervised link prediction problem in a multi-relational network from an OHC. Overall, the results show that considering different types of relationships in a multi-relational social network can improve the performance of link predictions in this context. In addition, we showed that community structures, as well as nodal similarities based on embedding, could further enhance the performance of our prediction.

The results have important implications for the design and management of an OHC. An OHC can develop a recommender system that recommends contents to users so that they read others' blog posts or participate in group discussions. The OHC design may even be improved by recommending

friends so that users would be able to access other users' wall or even let users send direct messages to each other if they belong to the same community. The more the users get connected within the community, the richer the information they would share and receive from each other. As other OHC related studies show that better engagement of users in OHC helps them to achieve their goals, such as reduce weight [9], recover their symptoms [11], or quit smoking [10], we expect these recommendations would help people at BAX to quit smoking.

Similar logic applies to users that have high nodal similarity based on the result of embedding. The users that have high similarity score in embedding may have similar characteristic but would have been detected for different communities. By allowing these users to communicate with each other in one of the four channels, users may benefit from each other by receiving information outside of the community. Having this functionality in OHC would possibly lead to a more interconnected network that is desirable since each user would be exposed to the breadth of information.

There are also interesting future research directions. For example, all of the experiments are based on undirected and unweighted networks. In the future, we may experiment with directionality between users and the frequency of the user interaction as to their implications in link prediction. Moreover, we plan to apply our methods to networks in other domains to see if our method generalizes to other networks.

References

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[2] François Chollet et al. Keras. `https://keras.io`, 2015.

[3] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review E*, 70(6):066111, 2004.

[4] Gennaro Cordasco and Luisa Gargano. Community detection via semi-synchronous label propagation algorithms. In *2010 IEEE International Workshop on: Business Applications of Social Network Analysis (BASNA)*, pages 1–8. IEEE, 2010.

[5] Peng Cui, Xiao Wang, Jian Pei, and Wenwu Zhu. A survey on network embedding. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[6] Darcy Davis, Ryan Lichtenwalter, and Nitesh V Chawla. Multi-relational link prediction in heterogeneous information networks. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 281–288. IEEE, 2011.

[7] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[8] Linyuan Lü and Tao Zhou. Link prediction in complex networks: A survey. *Physica A: statistical mechanics and its applications*, 390(6):1150–1170, 2011.

[9] Xiaoxiao Ma, Guanling Chen, and Juntao Xiao. Analysis of an online health social network. In *Proceedings of the 1st ACM international health informatics symposium*, pages 297–306. ACM, 2010.

[10] Sahiti Myneni, Kayo Fujimoto, Nathan Cobb, and Trevor Cohen. Content-driven analysis of an online community for smoking cessation: integration of qualitative techniques, automated text analysis, and affiliation networks. *American journal of public health*, 105(6):1206–1212, 2015.

[11] Priya Nambisan. Information seeking and social support in online health communities: impact on patients' perceived empathy. *Journal of the American Medical Informatics Association*, 18(3):298–304, 2011.

[12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[13] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

[14] Lei Tang and Huan Liu. Scalable learning of collective behavior based on sparse social dimensions. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1107–1116. ACM, 2009.

[15] Xi Wang and Gita Sukthankar. Link prediction in multi-relational collaboration networks. In *Proceedings of the 2013 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 1445–1447. ACM, 2013.

[16] Yang Yang, Nitesh Chawla, Yizhou Sun, and Jiawei Hani. Predicting links in multi-relational and heterogeneous networks. In *2012 IEEE 12th international conference on data mining*, pages 755–764. IEEE, 2012.

[17] Colleen Young. Community management that works: how to build and sustain a thriving online health community. *Journal of medical Internet research*, 15(6):e119, 2013.

[18] Kang Zhao, John Yen, Louis-Marie Ngamassi, Carleen Maitland, and Andrea H Tapia. Simulating inter-organizational collaboration network: a multi-relational and event-based approach. *Simulation*, 88(5):617–633, 2012.