Infer User Interests via Link Structure Regularization

Jinpeng Wang, [1]State Key Laboratory of Software Development Environment at Beihang University, [2]Department of Computer Science at Peking University
Wayne Xin Zhao, Department of Computer Science, Peking University
Yulan He, School of Engineering and Applied Science, Aston University
Xiaoming Li, [1]State Key Laboratory of Software Development Environment at Beihang University, [2]Department of Computer Science at Peking University

Learning user interests from online social networks helps to better understand user behaviors and provides useful guidance to design user-centric applications. Apart from analyzing users' online content, it is also important to consider users' social connections in the social web. Graph regularization methods have been widely used in various text mining tasks, which can leverage the graph structure information extracted from data. Previously, graph regularization methods operate under the cluster assumption that nearby nodes are more similar and nodes on the same structure (typically referred to as a cluster or a manifold) are likely to be similar. We argue that learning user interests from complex, sparse and dynamic social networks should be based on the link structure assumption under which node similarities are evaluated based on the local link structures instead of explicit links between two nodes. We propose a regularization framework based on the relation bipartite graph, which can be constructed from any types of relations. Using Twitter as our case study, we evaluate our proposed framework from social networks built from the retweet relations. Both quantitative and qualitative experiments show that our proposed method outperforms a few competitive baselines in learning user interests over a set of predefined topics. It also gives superior results compared to the baselines on retweet prediction and topical authority identification.

Categories and Subject Descriptors: H.4 [**Information Systems Applications**]: Miscellaneous

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: User interests, graph regularization, link structure

ACM Reference Format:
Jinpeng Wang, Wayne Xin Zhao, Yulan He, Xiaoming Li, 2013. Infer User Interests via Link Structure Regularization. ACM Trans. Intell. Syst. Technol. V, N, Article A (January YYYY), 22 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

With the growing popularity of social media tools such as Twitter and Facebook, millions of users actively participate in these social media platforms and engage in online social activities, e.g., posting a tweet and uploading a photo. User interests can be manifested through contents shared in a social context and various types of social activities. Learning user interests from social networks is particularly useful to understand online user behaviors and has thus attracted much research attention in recent

Submission date:30-Oct-2012; Revision date:22-Apr-2013; Acceptance date:16-Jun-2013.
The first two authors contributed equally to this work and should be considered as joint first authors.
Authors addresses: J. Wang, Peking University and Beihang University; X. Zhao (corresponding author), Peking University, email: batmanfly@gmail.com; Y. He, Aston University; X. Li, Peking University and Beihang University.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

years [Han et al. 2012; Guo et al. 2009; Ahmed et al. 2011; Wang et al. 2011; Shi et al. 2009]. Apart from analyzing the shared contents in the social network such as users' tweets or blog entries, it is also important to consider the online social network connections [Tang et al. 2009; Weng et al. 2010; Bakshy et al. 2011; Anagnostopoulos et al. 2008]. For example, people find someone interesting in Twitter, and 'follow' them to subscribe to their tweets. They may also be followed by others. Through such social connections, user interests might be driven to be similar.

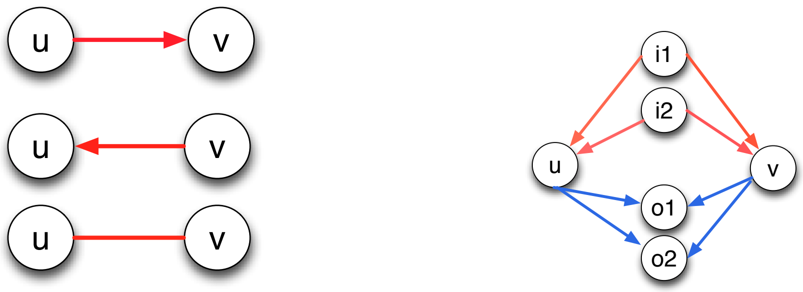
Social connections are often represented by graphs where nodes are users and links indicate two users are connected. To leverage graph links, cluster assumption is often used that nearby nodes are more similar and nodes on the same structure (typically referred to as a cluster or a manifold) are likely to be similar [Zhou et al. 2004]. Various studies adopt this assumption in different tasks. In classification, nearby nodes or nodes in the same structure have similar labels [Li et al. 2008; Ji et al. 2011]. In learning topic similarities, similar nodes have close topic distributions [Mei et al. 2008]. In online social networks, this assumption can be explained as an online user having similar interests with her neighbors or friends [Ma et al. 2011], for example, followees on Twitter or friends in Facebook.

Online social networks are complex in nature, and there can be multiple types of social connections between users. For example, in Twitter, there are three major types of social links between two users [Kwak et al. 2010; Welch et al. 2011]: (1) *following*, a user has added another user in her friend list; (2) *retweeting*, a user has forwarded a tweet from another user; (3) *mentioning*, a user has included another user in her own tweet. As shown in previous studies, these links may indicate different levels of topical relevance, e.g., retweeting is a stronger indicator of topical relevance than following [Welch et al. 2011]. As such, algorithms based on the cluster assumption may not be effective here since they rely on a single link between two nodes and do not distinguish different types of social interactions. Also, social networks built on various types of social interactions are very sparse and dynamic in nature. Take Twitter as an example: (1) about 42% of users have fewer than five followers [1]; (2) link changes can happen where existing following links might be removed and new links can be added [Hopcroft et al. 2011]. Therefore simple application of algorithms based on the cluster assumption to online social networks may not lead to good performance.

We argue that learning user interests from complex, sparse and dynamic social networks should be based on the *link structure assumption*. In particular, under the *link structure assumption*, node similarities are not measured based on the existence of explicit links. Instead, they are evaluated based on the local link structures between two nodes. For example, people sharing many followers or followees are likely to be similar in terms of their topical interests. Hence, nodes with similar local link structures tend to be similar. Compared to the traditional cluster assumption, the link structure assumption can be potentially more robust to adapt to various types of social connections and more resilient to sparse and dynamic networks. An illustrative example to compare these two assumptions is shown in Figure 1.

In this article, we propose a novel algorithm of learning user interests from social networks based on the *link structure assumption*. User interests can be reflected in various forms, e.g., interests over items in rating systems [Ma et al. 2009b] or interests over trending topics in microblogs [Zhao et al. 2011]. We consider user interests as distributions over topics in Twitter, i.e., we would like to learn the relative weights which measure interests on hot topics for a user. Here a topic refers to a semantically coherent theme which receives substantial attention from users, e.g., "Health Care Reform" and "Iran Election".

[1] The statistics are obtained on the data set in [Kwak et al. 2010].



(a) Traditional cluster assumption.

(b) Link structure assumption.

Fig. 1. An example to compare the two assumptions. The traditional cluster assumption assumes that two nodes are similar if there is an explicit link between them and then node values can be regularized through the links; The link structure assumption examines the local link structure of nodes, e.g., the sharing of common in-links and out-links. Here, u, v, i_1, i_2, o_1, o_2 are nodes in a graph, u, v share common in-links from i_1, i_2 and out-links to o_1, o_2.

To model the link structure assumption, we propose a regularization-based framework by utilizing the *relation bipartite graph*, which can be constructed based on any types of relationships. Take Twitter as an example, the retweet links can be used to build a relation bipartite graph. Our framework consists of two regularization factors, out-link regularization and in-link regularization, which naturally transform directed relationships into undirected relationships. We perform both quantitative and qualitative evaluations on the Twitter data set. We show that our method outperforms a baseline which does not consider social network connections. It also gives superior performance compared to a method based on the traditional cluster assumption.

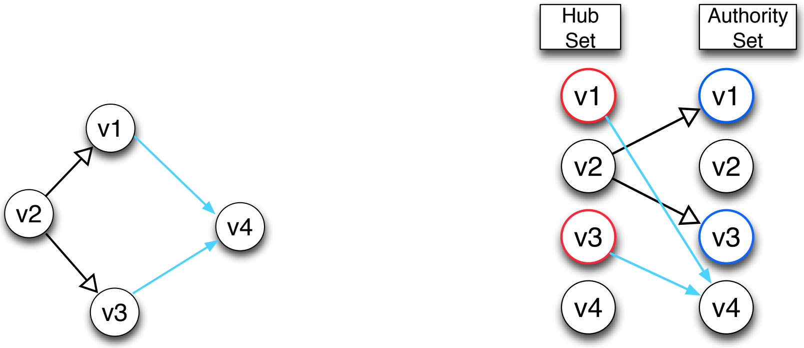
To the best of our knowledge, no prior work has studied the link structure assumption in online social networks. With the growing popularity of online social networks there is a urgent need to consider various types of social connections for analyzing online content and understanding user behaviors. Our proposed framework provides a principled solution to model social connections, especially directed social relations.

2. PROBLEM DEFINITION

We study the problem of inferring user interests over *topics* in Twitter. User interests over various topics in Twitter can be manifested in a number of different ways, including reading tweets, following hot topics, forwarding tweets from friends (a.k.a. retweet) and publishing original tweets. We focus on one particular activity that a user posts tweets on her interested topics. A tweet is a short document with a limit of 140 characters. We do not discriminate between *retweets* and *original tweets*.

Topic: A *topic* t is a semantically coherent theme which receives substantial attention from users. Here we do not make any specific assumption on topic representations. A topic can be represented as a multinomial distribution over the vocabulary, a trending hashtag, or a keyphrase of named entities, etc. Let $\mathcal{T}$ be the set of K hot topics in Twitter.

Relation Graph: Let $\mathcal{U}$ be the set of users in Twitter. Formally, a network of users for relation R can be defined as a directed graph $\mathcal{G}_R = (\mathcal{U}, \mathcal{E}_R)$, and the statement that there exists relation R between two users u and v can be denoted as an ordered pair $(u, v) \in \mathcal{E}_R$. The direction of the edge (i.e., $u \to v$) corresponds to the direction of the relation. E.g., if we instantiate R to be *retweet*, $u \to v$ can indicate that u has retweeted



(a) Relation graph.

(b) Relation bipartite graph.

Fig. 2. An example to illustrate the transformation from a relation graph to a relation bipartite graph. There are in total four vertices. For easy visualisation, we use different types of arrows to indicate in-links and out-links for $\{v_1,v_3\}$ respectively.

at least one message from v. Given an edge $u \to v$, v may potentially influence the interests of u.

Relation Bipartite Graph: From the relation graph $\mathcal{G}_R$, we can construct the relation bipartite graph $\mathcal{G}'_R = (\mathcal{V}_H, \mathcal{V}_A, \mathcal{E}_R)$ for relation R. Here we define $\mathcal{V}_H, \mathcal{V}_A$ to be exactly the same as $\mathcal{U}$. Given a pair of users (h, a), $a \in \mathcal{V}_A, h \in \mathcal{V}_H$, we build a link from a to h if there is a link between a and h in the original relation graph. Further, we define a weight $w_R(h, a)$ to be associated with this link, which indicates the strength between h and a in relation R. An example of how to derive a relation bipartite graph from a relation graph is illustrated in Figure 2. Based on this definition, we can see that there are two different roles for users in Twitter in analogy with the HITS algorithm, hubs and authorities.

With the *relation* defined here, it can be seen that any type of directed connection in online social networks can be modeled by the relation bipartite graph. For a reciprocal link or undirected link between two nodes u and v, we can add two directed links $u \to v$ and $v \to u$ to the relation bipartite graph.

Interest profile: Given a time span $[s_b, s_e]$, an interest profile of a user u is represented by a vector of weights $f_u(\cdot)$ over K topics, i.e., $(f_u(1), ..., f_u(K))$, where $f_u(i)$ is a weight which measures the degree of u's interests in ith topic. We further require that $\sum_i f_u(i) = 1, f_u(i) \geq 0, \forall u \in U$.

The main goal of our task is to infer the interest profile over topics for each user in U in a time interval $S = [s_b, s_e]$ based on the historical data we have. In this article, we mainly focus on two types of topics: (1) multinomial topics, i.e., topics learnt from topic models such as Latent Dirichlet Allocation (LDA) [Blei et al. 2003]; (2) hashtag topics, i.e., a hashtag itself is treated as a topic.[2] The first type is widely studied in topic modeling from text; the second type is mainly rooted in microblogs, which is designed to better search and organize information. Although topics can evolve in a complex manner (e.g., emerging, growing, shrinking, etc.), we simplify the problem by making an assumption here that topics in S are fixed and independent[3]. In practice,

[2] It is worth noting that topic extraction from online social networks is not the focus of the article. Hence, we simply assume that topics can either be extracted using existing methods (such as LDA), or are simply given (such as hashtag topics).

[3] This is mainly for the ease of evaluation since it is difficult to evaluate user interests over bursty topics.

the predicted interval S is usually short and hence most topics are persistent within S.

3. THE NAÏVE MAXIMUM LIKELIHOOD ESTIMATION

In this section, we present a simple method which considers each user independently without taking into account social networks and discuss how to estimate user interests based on the historical data which contains topic-related content that users write or retweet. Given a set of K topics and a user u, we first count the number of activities u performs over these K topics, i.e., $n_{u,1}, ..., n_{u,K}$. Then we apply the maximum likelihood method to estimate the underlying interest profile $(f_u(1), ..., f_u(K))$. Formally, we can write the likelihood function

$$\mathcal{L}_M = \prod_i f_u(k)^{n_{u,k}}.$$

To maximize this function, we set $f_u(k) = \frac{n_{u,k}}{\sum_{k'} n_{u,k'}}$. In practice, we use the additive smoothing to avoid "dividing by zero",

$$f_u\hat{(}k) = \frac{n_{u,k} + \alpha}{\sum_{k'=1}^{K} n_{u,k'} + K\alpha}, \tag{1}$$

where $\alpha > 0$ controls the smoothing degree. We refer to $\hat{f}_u(i)$ as the naïve estimation of $f_u(i)$ since it only considers each user in isolation without taking into account the attached social networks.

For multinomial topics, we set $n_{u,k}$ to be the number of tokens which is assigned to topic k for user u, and set $\alpha = \frac{50}{K}$ as in LDA [Griffiths and Steyvers 2004]; for hashtag topics, we set $n_{u,k}$ to be the number of the kth hashtag that user u has used in her tweets, and set α to 1, which is essentially the add-one smoothing.

Although this method seems very straightforward, it has a few shortcomings. Firstly, merely relying on users' own historical data may not be able to capture user interests completely. Secondly, users' own historical data can be very sparse, therefore ML estimation often fails to derive users' interest profiles accurately. Finally, as showed in [Weng et al. 2010], users with the same set of following links are more similar than those without. Hence, social connections should be taken into account when deriving users' interest profiles.

4. OUR MODEL

The aforementioned method oversimplifies the complicated nature of user interests. We propose a novel framework to model user interests based on the link structure assumption that users sharing many common in-links (i.e. parent links) or out-links (i.e. child links) in a relation bipartite graph should have similar topic interests.

4.1. The regularization framework

Our proposed framework consists of three factors, out-link regularization, in-link regularization, and a fitting constraint.

Out-Link regularization: Two users are similar if they share many out-links in a relation bipartite graph, i.e., they co-link many common *authorities*. In Figure 2(b), both v_1 and v_3 in the hub set link v_4 in the authority set, so the values of v_1 and v_3 tend to be correlated.

The out-link regularization factor tries to capture the similarity between vertices in the hub set $\mathcal{V}_H$ that are strongly related. We measure the relatedness between two vertices u and v with a function $w_A(u, v)$. The subscript of A in $w_A(u, v)$ denotes that the relatedness is computed based on the vertices in the authority set. Formally, we represent each user as a vector of authority weights, and then use the cosine function to compute the similarity

$$w_A(u,v) \;=\; \sum_{a\in\mathcal{V}_A} \frac{w_R(u,a)w_R(v,a)}{\sqrt{(\sum_{a'} w_R^2(u,a'))(\sum_{a'} w_R^2(v,a'))}}, \tag{2}$$

where $w_R(u,a)$ is the edge weight in the relation bipartite graph introduced in Section 2, which reflects the strength between u and a in relation R.

With the introduction of $w_A(u,v)$, we implement the out-link regularization factor

$$\Omega_A(f) \;=\; \frac{1}{2}\sum_{u,v\in\mathcal{U}} w_A(u,v)\sum_k (f_u(k)-f_v(k))^2. \tag{3}$$

In-Link Regularization: Two users are similar if they share many common in-links in a relation bipartite graph, i.e., they are linked by many common *hubs*. In Figure 2(b), both v_1 and v_3 in the authority set are linked by v_2 in the hub set. Thus the values of v_1 and v_3 tend to be correlated.

The in-link regularization factor tries to capture the similarity between vertices in the authority set $\mathcal{V}_A$ that are strongly related. We measure the relatedness between two vertices u and v with a function $w_H(u,v)$. The subscript of H in $w_H(u,v)$ denotes that the links are computed based on the vertices in the hub set. Similar to $w_A(u,v)$, we have

$$w_H(u,v) \;=\; \sum_{h\in\mathcal{V}_H} \frac{w_R(h,u)w_R(h,v)}{\sqrt{(\sum_{h'} w_R^2(h',u))(\sum_{h'} w_R^2(h',v))}}. \tag{4}$$

With the introduction of $w_H(u,v)$, we implement the in-link regularization factor

$$\Omega_H(f) \;=\; \frac{1}{2}\sum_{u,v\in\mathcal{U}} w_H(u,v)\sum_k (f_u(k)-f_v(k))^2. \tag{5}$$

Fitting Constriant: The third factor we consider is that user interests learnt from our models should not deviate too much from the interests estimated from users' own data, i.e., $\hat{f}_u(k)$. Formally, we define the fitting constraints

$$\Omega_F(f) = \frac{1}{2}\sum_u\sum_k (f_u(k)-\hat{f}_u(k))^2. \tag{6}$$

Combining Equations 3, 5 and 6, our objective function is a linear combination of these three cost functions:

$$\mathcal{O}(L) = \alpha\times\Omega_A(f) + \beta\times\Omega_H(f) + \gamma\times\Omega_F(f), \tag{7}$$

where $\alpha+\beta+\gamma=1$ and $\alpha,\beta,\gamma\geq 0$. We can tune α,β,γ based on different data sets. This model considers two aspects of link structures, namely in-links and out-links, and **we denote it as CoReg**.

This general framework can capture various relations through the two regularization factors introduced in Equations 3 and 5. Intuitively, to minimize our objective function, we have to seek a trade-off between information derived from users' own historical data and information from similar users. Usually, the values of α, β and γ can be set either empirically or by incorporating prior knowledge. The relation strength $w_R(\cdot,\cdot)$ can be set differently for different relations. We observe in Figure 2 is that there is no explicit link between v_1 and v_3. Using CoReg, we can add two virtual links between them, one is through the authority vertex v_4 and the other is through the hub vertex v_2.

A previous study [Ma et al. 2011] utilizes the friend-like relationships, e.g., the following relation or the trust relation, by examining whether there is an explicit link between two users. However, not all relations indicate friendship and thus some of these relations may have weak topical similarities. As a result, regularization based on a single link between two users lacks the robustness in capturing user interests. Different from [Ma et al. 2011], our proposed framework examines the similarity between the local structures of two vertices, which is potentially more robust in learning user interests. Also, through modeling two different roles for vertices, hub and authority, our framework naturally transforms directed relations into undirected ones.

4.2. Model Learning

We can separate different cost functions to obtain the partial derivatives with respect to $f_u(k)$ as follows

$$\begin{aligned}
\frac{\partial \Omega_A(f)}{\partial f}|_{u,k} &= w_A(u,v)(f_u(k) - f_v(k)),\\
\frac{\partial \Omega_H(f)}{\partial f}|_{u,k} &= w_H(u,v)(f_u(k) - f_v(k)),\\
\frac{\partial \Omega_F(f)}{\partial f}|_{u,k} &= f_u(k) - \hat{f}_u(k).
\end{aligned}$$

The derivative of Equation 7 is a linear combination of the above derivative functions

$$\frac{\partial \mathcal{O}(L)}{\partial f}|_{u,k} = \alpha \times \frac{\partial \Omega_A(f)}{\partial f}|_{u,k} + \beta \times \frac{\partial \Omega_H(f)}{\partial f}|_{u,k} + \gamma \times \frac{\partial \Omega_F(f)}{\partial f}|_{u,k}.$$

By setting $\frac{\partial \mathcal{O}(L)}{\partial f}|_{u,k}$ to zero, we get an iterative formula to derive $f_u(k)$ as follows

$$f_u(k) = \frac{\alpha \sum_v w_A(u,v) f_v(k) + \beta \sum_v w_H(u,v) f_v(k) + \gamma \hat{f}_u(k)}{\alpha \sum_v w_A(u,v) + \beta \sum_v w_H(u,v) + \gamma}. \tag{8}$$

It can be easily verified that this equation satisfies:

$$\forall u, \sum_u f_u(k) = 1 \text{ and } f_u(k) \geq 0.$$

We present a pseudo code for the proposed method in Algorithm 1. Here, $iter$ is a predefined number of iterations. Steps 1-6 are for initialization and steps 7-18 for the iterative algorithm. We discuss some of the implementation details below. Storing the matrix $w_A(\cdot,\cdot)$ usually takes up significant memory space. In practice, we do not store all the edge weights but only keep the edges with large weights to allow more efficient computation in Step 10 in Algorithm 1.

4.3. Discussion of the model

Variants and connections to existing methods.. Our proposed optimization framework can be generalized to many different tasks. In this subsection, we make connections with

ALGORITHM 1: Algorithm for learning user interests.

```
for $u \in \mathcal{U}$ do
    for $k = 1$ To $K$ do
        Set $\hat{f}_u(k)$ according to the simple MLE in Equation 1;
        Set $f_u(k) = \frac{1}{K}$;
    end
end
for $n = 1$ To $iter$ do
    for $u \in \mathcal{U}$ do
        for $k = 1$ To $K$ do
            learn $f_u(k)$ according to Equation 8;
        end
    end
    if $n > 1$ then
        if $\sum_k (f_u^{(n)}(k) - f_u^{(n-1)}(k))^2 \leq 1e-5$ for all $u$ then
            break;
        end
    end
end
```

other existing methods. There are three parameters in our objective function, α, β and γ, which represent the weights of different constraints on the relation bipartite graph. Different settings of these parameters lead to several variants:

(1) $\alpha \neq 0, \beta \neq 0, \gamma \neq 0$. Equation 8 can be re-written as follows

$$f_u(k) \ = \ \frac{\gamma}{C_u}\hat{f}_u(k) + \sum_v \frac{\alpha \times w_A(u,v) + \beta \times w_H(u,v)}{C_u} f_v(k),$$

where $C_u = \alpha \sum_v w_A(u,v) + \beta \sum_v w_H(u,v) + \gamma$. This is similar to PageRank except that we have a vertex-specific weight of $\frac{\gamma}{C_u}$ to invoke the restart. We can see that $f_u(k)$ has a fraction of $\frac{\gamma}{C_u}$ to retain its naïve estimation. We can also assume the existence of a pseudo edge between u and v, and the edge weight is $(\frac{\alpha \times w_A(u,v) + \beta \times w_H(u,v)}{C_u})$, which propagates v's evidence to u.

(2) $\alpha = 0, \beta \neq 0, \gamma \neq 0$. Only the regularization through hub vertices is considered.

(3) $\alpha \neq 0, \beta = 0, \gamma \neq 0$. Only the regularization through authority vertices is considered.

(4) $\alpha = 0, \beta = 0, \gamma = 1$. Our method reduces to the simple MLE method, which ignores users' social networks.

(5) $\alpha \neq 0, \beta \neq 0, \gamma = 0$. In this case, our method is built fully on the social networks without evidence from users themselves.

Detailed results obtained using our proposed framework with different parameter settings will be presented in Section 6.7.

Reexamination of the link structure assumption.. Finally, we re-examine the link structure assumption described in Section 1. By studying Equations 3 and 5 carefully, it can be observed that the traditional cluster assumption is actually embedded in the link structure assumption. Given two vertices u and v, two virtual links can be built between them with weights of $w_H(u,v)$ and $w_A(u,v)$ respectively. With such virtual links, vertices similarity can be measured based on the cluster assumption. Although the link structure assumption can be reduced to the cluster assumption with the proper insertion of virtual links, the former is more general than the latter since it also takes into account local link structure apart from explicit links between two nodes when evaluating node similarities.

5. INSTANTIATION OF OUR METHOD

Our framework can account for different types of relations. In this section, we show an application of our method by considering a specific type of relation between Twitter users, the retweeting relation. Compared with the following relation, the retweeting relation is a much stronger indicator of social influence [Welch et al. 2011].

We first set the relation strength $w_R(u, v)$ between u and v to the number of tweets that u has forwarded to v. The in-link and out-link regularizations can be explained below:

Out-link $\mapsto$ **Co-Retweet**: Two users are similar if they constantly forward the tweets of common "authorities". For example, two users who often forwarded tweets from @cnnbrk and @nytimes are likely to have similar interests in news related topics. Similarly, two users who often forwarded tweets from music celebrities are likely to have similar interests in music related topics. We let $w_A(u, v)$ be the co-retweet similarity between u and v. The subscript of A in $w_A(u, v)$ denotes that the links are built through the "authorities" that u and v co-retweet. We can compute $w_A(\cdot, \cdot)$ according to Equation 2.

In-link $\mapsto$ **Co-Retweeted**: Two users are similar if their tweets are often forwarded by common "hubs". For example, both Lady Gaga and Justin Bieber are popular music celebrities, and it is common to see many Twitter users are fans of both of them and therefore could re-tweet many of their tweets. Similar to $w_A(\cdot, \cdot)$, we let $w_H(u, v)$ be the co-retweeted similarity between u and v. The subscript of H in $w_H(u, v)$ denotes that the links are built through the "hubs" who co-retweet u and v. We can compute $w_H(\cdot, \cdot)$ according to Equation 4.

With the instantiation of $w_H(\cdot, \cdot)$ and $w_A(\cdot, \cdot)$, we then run Algorithm 1 iteratively until it converges. Besides the methods in Equations 2 and 4, we can also use other similarity measurement methods, e.g., Jaccard similarity, to set $w_A(\cdot, \cdot)$ and $w_H(\cdot, \cdot)$. Nevertheless, they don't work better compared to our proposed similarity measurement method in our experiments. We have also tested our framework using the *following* relation. The conclusions drawn are similar to those using the *retweet* relation. Due to the space limit, in this article we only report the experimental results obtained using the *retweet* relation.

6. EXPERIMENTS AND RESULTS

6.1. Construction of test collection

We evaluate our method on a Twitter data set which spans the second half of year 2009 [Kwak et al. 2010]. We take the data in August and September 2009 for evaluation.

Since our focus is to study how to leverage various social connections to learn user interests, we simplify this evaluation task as follows: (1) we only consider persistent topics, so that user interests over them should be relatively stable; (2) although relation networks can change in these two months, we take a snapshot of relation networks at the end of September 2009 and then use it in all our experiments. We use the data in August 2009 for training and the data in September 2009 for testing. Training and test data with smaller sizes, e.g., two weeks, have also been evaluated. However, the resulting data for a single user becomes very sparse, and hence the evaluation results are less meaningful.

We consider two types of topical representations. The first type is the topics learnt from topic models such as LDA [Blei et al. 2003], where each topic is represented as a multinomial distribution over the terms in the vocabulary. Similar to the finding in [Zhao et al. 2011], we find that most of the topics generated by LDA are long-standing topics. There are a few event related topics such as "Health Care Reform", "Iran Elec-

Table I. Statistics of our data set.

#users	28,825
#tweets	19,067,877
#follow-link	1,271,472
#retweet-link	415,089

tion" and "Afghanistan War", which are also persistent topics. The second type is the trending topics, which are the most popular themes discussed in Twitter. We select the most frequent and relatively stable hashtags in Twitter as follows. We consider a hashtag to be daily active if at least ten different users used it in a day. We only keep the hashtags which are active for at least ten days in both the training and test data. We refer to topics of the first type as *multinomial topics*, and the second type as *hashtag topics*.

In Twitter, a large number of users do not use hashtags at all. As such, if we simply select users randomly, we may end up with the case where most of the sampled users rarely used any hashtags. In order to avoid such a problem, we first select the top 60,000 users which used hashtags most in their tweets. We further divide all the users into six groups and randomly select 5 users in each group to form a set of 30 seed users. Then we use a breadth-first search to add users by following the retweeting links of these seed users, and run the search algorithm with two iterations. Finally, we get 28,825 users. We keep all the tweets from these users and build the retweet graph among them. The same data set is used for the evaluation of both multinomial topics and hashtag topics. We want to examine the performance of our proposed framework for different types of topics on the same set of users. We summarize our data statistics in Table I.

6.2. Methods to compare

We consider the following baseline methods for comparison:

MLE: The simple maximum likelihood estimation has been described in Section 3, in which we do not consider the social networks. Users' interest profiles are derived using Equation 1.

InfPR: We also consider a simplified version of the method in [Ma et al. 2011] which is based on the cluster assumption that a user should be similar to her "friends".

$$\arg\min_{\{f_u(\cdot)\}} \ \gamma \times \frac{1}{2}\sum_u \sum_k (f_u(k) - \hat{f}_u(k))^2 + \\ (1-\gamma) \times \frac{1}{2} \sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{R}_u} w(v,u) \sum_k (f_u(k) - f_v(k))^2,$$

where $w(v,u)$ is the normalized number of tweets that u has retweeted from v, $\mathcal{R}_u$ is the set of users that u has retweeted from, and γ is a coefficient to tune.

After some derivations, the following iterative equation can be obtained

$$f_u(k) = (1-\gamma) \sum_{v \in \mathcal{R}_u} w(v,u) f_v(k) + \gamma \hat{f}_u(k). \tag{9}$$

Since it is similar to PageRank, we denote it as influence PageRank (InfPR).

CoReg: This is our proposed method as described in Section 4.

OutReg: A variant of our proposed method by setting $\beta = 0$ in Equation 8, i.e., we only consider out-link regularization.

InReg: Another variant of our proposed method by setting $\alpha = 0$ in Equation 8, i.e., we only consider in-link regularization.

Apart from MLE, all the other methods run iteratively until convergence or until they reach the maximum number of iterations which is empirically set to 10 here.

6.3. Evaluation on multinomial topics

For evaluation on multinomial topics, we first learn topics using standard topic models such as LDA, and then estimate user interests, i.e., the users' topic distributions, over these topics. For each user, we generate two documents with the first one containing her tweets in August 2009 as the training document and the second one containing her tweets in September 2009 as the held-out document. Then we use the implementation of LDA in MALLET[4] and train it on all the training documents. LDA generates a set of topics $\{\theta_k\}$ and a unique topic distribution for a user u, $\{\phi_u(k)\}$ for August 2009. Let $\{\phi_u(k)\}$ be the assignment of $\{\hat{f}_u(k)\}$ in CoReg and InfPR. After running these algorithms, we can obtain multiple estimations of user interests, which are *probability distributions*, from different methods.

This evaluation task aims to examine the overall generalization ability of modeling unseen or held-out data. The commonly used perplexity measure is adopted as the evaluation metrics of document modeling. A lower perplexity score indicates better generalization or prediction performance [Blei et al. 2003]. In our experiments, a test "document" consists of all the tweets posted by a user in September 2009. Given a test set $\mathcal{D}_{test}$, the perplexity is computed as:

$$\begin{aligned} perplexity(\mathcal{D}_{test}) &= \exp\left\{ - \frac{\sum_{d_u \in \mathcal{D}_{test}} \log P(\mathbf{w}_{d_u})}{\sum_{d_u \in \mathcal{D}_{test}} N_d} \right\}, \\ &= \exp\left\{ - \frac{\sum_{d_u \in \mathcal{D}_{test}} \sum_{i=1}^{N_{d_u}} \log P(w_{d_u,i})}{\sum_{d_u \in \mathcal{D}_{test}} N_{d_u}} \right\}, \\ &= \exp\left\{ - \frac{\sum_{d_u \in \mathcal{D}_{test}} \sum_{i=1}^{N_{d_u}} \log \Big(\sum_{k=1}^{K} P(w_{d_u,i}|\theta_k) f_u(k) \Big)}{\sum_{d_u \in \mathcal{D}_{test}} N_{d_u}} \right\}, \end{aligned}$$

where d_u is a document from user u in $\mathcal{D}_{test}$, $\mathbf{w}_{d_u}$ is the token stream of d_u, N_{d_u} is the number of tokens in d_u, K is the number of topics and $P(w_{d_u,i}|\theta_k)$ is the probability of word $w_{d_u,i}$ given the kth topic. A better method should yield a smaller perplexity value on the held-out document set. It is worth noting here that by explicitly introducing the user interests $f_u(k)$, perplexity is actually calculated in a similar way as that in the Author-Topic model [Steyvers et al. 2004]. In order to evaluate the impact of user interests, we treat each tweet as a document as opposed to the previous experiments where all the tweets of a user are concatenated into a single document. We then train LDA on the tweets in the training set and perform inference on the tweets in the test set to compute the perplexity results. We denote the results as "LDA" in Table II.

We empirically set $\alpha = \beta = 0.05, \gamma = 0.9$ in CoReg and $\gamma = 0.9$ in InfPR. The perplexity results are presented in Table II by varying the number of topics between 50 and 100. It can be observed that InfPR and CoReg outperforms MLE indicating the importance of incorporating social relationships for learning user interests. Both OutReg and InReg perform similarly and they give a better performance compared to InfPR. CoReg is a combination of OutReg and InReg, and it gives the best results compared to all the other baselines. The difference between CoReg and InfPR is that CoReg can

[4] http://mallet.cs.umass.edu

Table II. Comparisons on multinomial topics in perplexity. Lower value indicates better performance.

Methods	50	75	100
LDA	64354	63213	62934
MLE	42964	42905	42414
InfPR	41901	41778	41245
OutReg	40889	40771	40238
InReg	40874	40749	40215
CoReg	**39851**	**39719**	**39182**

leverage implicit relationships to better capture user interests, which is very important in sparse directed social networks. Finally, we notice that the incorporation of user interests significantly reduces the perplexities since all the baselines and our proposed methods outperform LDA by a large margin. This is inline with what has been observed in [Hong and Davison 2010] that the aggregation of tweets by users is more effective than treating each tweet as a separate document.

6.4. Evaluation on hashtag topics

The second type of topics we consider are hashtag topics. We select top 500 hashtags used in [Romero et al. 2011] since these hashtags receive substantial attention from Twitter users and have a broad coverage of topics. Inspired by the evaluation in information retrieval, we adopt precision@N as our evaluation metric. Equation 1 is used to initialize $\hat{f}_u(k)$ and the models are trained on the August 2009 data and tested on the September 2009 data. A candidate method will return a user interest profile which is a distribution over hashtags based on the training data, and the top N hashtags are compared against the actual hashtags used in the testing data for each user.[5] We compute the average precision of correctly predicted hashtags among the top N hashtags over all the users. In the Twitter data evaluated here, a user used 14 different hashtags on average (among our selected 500 hashtags). As such, N is set to 1, 3, 5, 7 and 10.

We empirically set $\alpha = \beta = 0.2, \gamma = 0.6$ in CoReg and $\gamma = 0.6$ in InfPR. The results are shown in Table III. Similar conclusions can be drawn as in the evaluation results on multinomial topics. Both InfPR and CoReg outperform MLE. CoReg and its two variants, OutReg and InReg, give better results compared to InfPR. Considering the in-link and out-link constraints simultaneously (CoReg) performs better than only taking into account one type of constraints (OutReg or InReg). We also notice that for recommending hashtags, α and β should be set with larger values compared with those for multinomial topics. The main reason is that for multinomial topics, users' interest profiles are estimated based on the word statistics over topics which are less sparse compared to the frequency of hashtag topics. Hence, for hashtag recommendation, statistics collected from a user's "similar" friends are boosted to alleviate the data sparsity problem.

After obtaining the user interests over the top N hashtags, we can further analyze the diversity of hashtag uses. We employ the entropy of user interests over hashtags to measure the diversity

[5] It is also possible to use the actual distributions of hashtags (calculated as the normalized occurrence frequencies of hashtags) for each user in the test data as ground truth and then calculate, for example KL divergence, between the learned user interest profiles and the actual hashtag distributions as evaluation metrics. Nevertheless, we argue that it makes more sense to predict the top N frequently used hashtags by a user based on the interest profile learned from her historical data.

Table III. Comparisons of precision@N for hashtags topics recommendation. Larger values indicate better performance. * denotes a significant improvement over the baselines.

Methods	P@1	P@3	P@5	P@7	P@10
MLE	0.606	0.512	0.495	0.523	0.512
InfPR	0.612	0.523	0.541	0.544	0.525
OutReg	0.630	0.531	0.577	0.565	0.543
InReg	0.634	0.533	0.570	0.554	0.530
CoReg	**0.650***	**0.559***	**0.580***	**0.571***	**0.554***

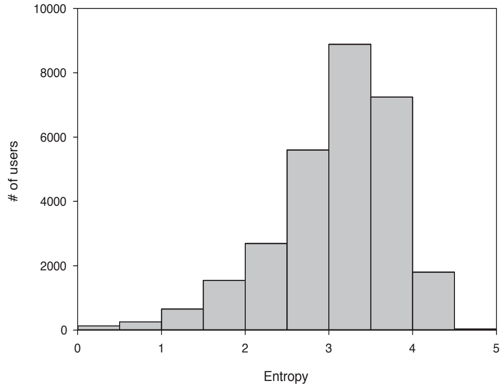


Fig. 3. Distribution of the user-level hashtag entropy.

$$\text{Entropy}(u) = -\sum_{k=1}^{K} f_u(k) \times \log f_u(k),$$

where $K = 500$ is the number of the most frequent hashtags considered here, $f_u(\cdot)$ are user interests learned from our method CoReg. A larger entropy indicates that a user tends to use more diverse hashtags; and a smaller value tells that a user only uses a small set of hashtags. We compute the hashtag entropy for every user in our data set and then rank users by their entropy values. Figure 3 shows that most users have an entropy value between 2.5 and 4. The “focused” users (≤ 1.5) are mostly organizations and groups. They use a very small set of hashtags to advertise or broadcast for themselves. The most “diverse” users are mainly common users who comment a lot on a wide range of topics and have at least several thousands followers. These users can be viewed as opinion leaders, who play a key role in the two-step information propagation as studied in [Wu et al. 2011].

6.5. Evaluation on retweet prediction

In this section, we further quantitatively evaluate different methods on retweet prediction which predicts whether a user will forward a tweet or not. Similar to the e-valuation on multinomial topics in Section 6.3, a set of topics $\{\theta_k\}$ and a unique topic distribution for a user u, $\{\phi_u(k)\}$, can be obtained from the training data in August 2009. Methods for learning user interests can then estimate $\{f_u(k)\}$ based on $\{\phi_u(k)\}$, which can be evaluated subsequently on the test data in September 2009.

Generally speaking, retweet prediction is a very challenging problem. Previous research [Hong et al. 2011; Feng and Wang 2013] explores the use of an extensive set

Table IV. Performance comparisons of retweet prediction.

Methods	MRR	P@10	P@20	P@30	P@100
MLE	0.199	0.060	0.114	0.163	0.406
InfPR	0.205	0.069	0.127	0.182	0.426
OutReg	0.214	0.073	0.131	0.187	0.432
InReg	0.213	0.072	0.130	0.187	0.431
CoReg	**0.234**	**0.084**	**0.146**	**0.207**	**0.456**

of features, including user features, style features, temporal features and content features for retweet prediction. Since our goal is to perform retweet prediction depending solely on user interests, we simplify the retweet prediction task as follows. For each user, we only consider the tweets of her followed users from whom she has forwarded at least one tweet in August 2009. We compute the topic similarity between a candidate tweet and the topical interest of a user. Then we rank these tweets based on the topic similarity scores in a descending order. A better method should be able to rank those tweets that the user has actually forwarded in higher positions.

Given a set of topic models $\{\theta_k\}_{k=1}^{K}$, we compute the conditional probability of the kth topic given a tweet d for each of topic

$$P(\theta_k|d) = \frac{\prod_{w \in d} P(w|\theta_k)}{\sum_{k'=1}^{K} \prod_{w \in d} P(w|\theta_{k'})}.$$

From which we derive the topic distribution for a tweet d, $\{P(k|d)\}_{k=1}^{K}$. Given a user and a set of candidate tweets, we first compute the negative KL-divergence of the topic distributions of the user and each of the candidate tweets, and subsequently rank these tweets in a descending order. We adopt precision@N and MRR (Mean Reciprocal Rank) commonly used in information retrieval as our evaluation metrics [Manning et al. 2008]. We conducted experiments with the topic number set to 50, 75 and 100 and found that the findings are similar regardless of the topic number settings. As such, we only report the results on 75 topics in Table IV.

The results shown in Table IV are consistent with what have been observed in Table II and III. MLE still performs the worst and incorporating social links improves upon MLE. Both OutReg and InReg give superior results than InfPR, and a combination of these two methods, i.e., CoReg, performs best. The overall performance of all the methods on retweet prediction is quite low since only user interest profiles are used for retweet prediction. Nevertheless, the results show the effectiveness of our proposed method over other baseline models.

6.6. Finding topical authorities

Finding topical authorities is an important text mining task in online social networks. We mainly consider two aspects to solve this problem, authority and topical relevance. We used the standard PageRank value of a user in the *retweet* graph as a measure of user authority, and used the learnt user interests as the measure of topical relevance. Formally, we compute the ranking score of user u on the kth topic according to the following equation

$$\text{score}(u, k) = \log \text{PageRank}(u) \times \theta_u(k),$$

where score(u, k) is the authority score of user u on the kth topic, PageRank(u) is the PageRank value of user u and $\theta_u(k)$ is the interest value on the kth topic learnt using CoReg or InfPR. Here we only consider hashtag topics as an illustration.

Table V. Top five topical authorities on the three example topics.

#charity		#starwars		#iran	
InfPR (#correct: 3)	CoReg (#correct: 5)	InfPR (#correct: 4)	CoReg (#correct: 5)	InfPR (#correct: 3)	CoReg (#correct: 5)
@globalgiving ✓	@globalgiving ✓	@star_wars_stuff ✓	@star_wars_stuff ✓	@iranwwp ✓	@dominiquerdr ✓
@hdrphotographer	@bonniegrrl ✓	@starwars_nns ✓	@starwars ✓	@iran_news ✓	@iranwwp ✓
@dcallejon	@ricklondon ✓	@bonniegrrl ✓	@bonniegrrl ✓	@whereismyvote_	@iran_news ✓
@contactafamily ✓	@lotay ✓	@johnhood ✓	@clubjade ✓	@eanewsfeed	@ikeoo ✓
@ricklondon ✓	@contactafamily ✓	@cgt2099	@johnhood ✓	@dominiquerdr ✓	@oxfordgirl ✓

The results were manually checked by looking into user accounts and the tweets they published. The judges took both relevance and authority into consideration. A topical authority should publish substantial content related to a given topic and meanwhile have a considerable number of followers. ✓ indicates a correctly identified topical authority.

We present the top five topical authorities on three example topics in Table V. We can see that CoReg has identified more correct topical authorities in the top five ranks than InfPR.

6.7. Further analysis of our regularized networks

Recall that our method is based on the link structure assumption which can be reduced to the traditional cluster assumption if we capture the implicit links by adding two virtual links between every set of two vertices with the weights of $w_A(\cdot,\cdot)$ and $w_A(\cdot,\cdot)$. In this section, we aim to shed some lights on the virtual links we constructed to understand why our method is effective. We select four graphs for comparison. The first graph is the retweet graph and used as the baseline for comparison. Our regularized networks are built through out-links and in-links, and the weight of a link between vertices u and v is set as $(\alpha w_A(u,v) + \beta w_H(u,v))$. We consider two variations based on CoReg by setting either α or β to zero. We refer to these two graphs as ***InReg*** and ***OutReg*** respectively.

We summarize the statistics of degree distribution in Table VI. On average, we can see that vertices in the InReg and OutReg networks have more neighboring vertices than those in the retweet graph. As a combination of InReg and OutReg, CoReg has the largest average node degree. We further plot the degree distribution of different networks in Figure 4. For the original retweet graph, we can see in Figure 4 that a large proportion of users have fewer than ten neighbors, and the influence propagation algorithms (e.g., InfPR) may not work well in such a sparse network. On the contrary, our CoReg can leverage more implicit relationships which are not captured in the retweet graph. By adding these virtual links between vertices, our regularized network is able to alleviate the network sparsity problem and hence gives more accurate results compared to other algorithms based on the cluster assumption.

Another interesting point is that InReg (the links built on co-retweeted) is much more sparse than OutReg (the links built on co-retweet). The main reason is that most of Twitter users usually receive very few retweets but tend to retweet more. Although InReg is relatively sparse compared to OutReg, it is able to capture the implicit relationships between top users in Twitter. For example, Lady Gaga and Justin Bieber don't usually retweet from each other or from common users. Hence there will be no links between these two stars in the retweet graph or the OutReg graph. On the contrary, it is common to see many users co-retweet from these two celebrities. This implied mutual interests in music shared between them. Such implicit relationships can be captured by the InReg graph.

We present four illustrative examples of implicit links found by our CoReg algorithm in Table VII. Users in the first two examples have the relation type of organization-member while users in the last two examples are linked because they published similar content and hence have similar topical interests. These implicit relationships are not captured by the retweet graph.

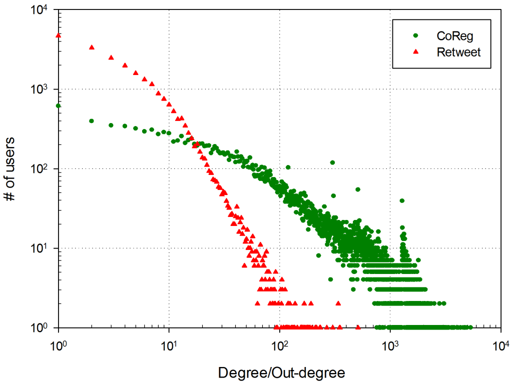


Fig. 4. Comparison of degree distributions (log-log scale). An out-link denotes that a user has forwarded tweets from another user.

Table VI. Comparisons of the average node degree for different networks built from our Twitter data set, a subset of the entire Twitter network. Note that the statistics are computed by keeping all the link edges for a node.

Retweet	OutReg	InReg	CoReg
14.40	115.16	59.88	158.88

Table VII. Examples of implicit links found by our CoReg algorithm.

User A	User B	Relationship Type
Jörg Tauss (@tauss)	Piratenpartei (@Piratenpartei)	Organization-Member
Explanations: Jorg Tauss is a German politician and former member of the Social Democratic Party of Germany (SPD). The Pirate Party Germany is a German political party founded in September 2006.		
Bonnie Burton(@bonniegrrl)	Star Wars(@starwars)	Organization-Member
Explanations: Star Wars is the official Twitter account of Lucasfilm StarWars.com. Bonnie Burton is a former content developer for Lucas Online, StarWars.com senior editor, staff writer for Star Wars Insidermagazine, and lead writer for the official star wars blog.		
Baratunde (@baratunde)	Liza Sabater (@blogdiva)	Users with similar interests
Explanations: Both Baratunde and Liza Sabater are famous political bloggers.		
Susan Cooper (@BuzzEdition)	Reg Saddler (@zaibatsu)	Users with similar interests
Explanations: Susan Cooper and Reg Saddler are social media enthusiasts.		

6.8. Practical consideration with temporal analysis

We have two kinds of weights for a pair of users u and v, namely $w_A(u, v)$ and $w_H(u, v)$. In practice, for a user u, we do not need to keep all edges of her linking neighbors, but only store the top M edges with the largest weights for $w_A(u, \cdot)$ or $w_H(u, \cdot)$ respectively. We find that $M = 30$ yields a good trade-off between computational efficiency and accuracy.

As we discussed before, due to the dynamic nature of the social networks, the weights $w_A(u, v)$ and $w_H(u, v)$ between a pair of users u and v tend to change over time. An important issue is how often we need to update these weights. Taking the retweet relation as an example, we split the data between August and September 2009 on a weekly basis and ended up with a total of 8 epochs. At the ith epoch, we use all the historical data up to the i epoch to learn weights $w_A(\cdot, \cdot)$ and $w_H(u, \cdot)$. Then for each user u, we can obtain her top M "closest" neighbors respectively for $w_A(u, \cdot)$ and $w_H(u, \cdot)$, denoted as a set $\mathcal{N}^{(i)}_{A,u}$ for $w_A(u, \cdot)$ and a set $\mathcal{N}^{(i)}_{H,u}$ for $w_H(u, \cdot)$, at the ith epoch. Two measures

Table VIII. Average change of the top 30 neighbors and their corresponding edge weights in terms of $w_A(\cdot,\cdot)$ by weeks.

Week	Consecutive change ($s = i - 1$)		Accumulative change ($s = 1$)	
(ith)	$\texttt{diff}^{(i,s)}$	$\Delta w^{(i,s)}$	$\texttt{diff}^{(i,s)}$	$\Delta w^{(i,s)}$
2	4.54	0.11	4.54	0.11
3	4.75	0.11	7.66	0.11
4	4.42	0.10	9.62	0.09
5	1.08	0.03	10.13	0.09
6	1.24	0.03	10.23	0.09
7	1.12	0.03	10.26	0.08
8	1.67	0.04	10.22	0.08

Table IX. Average change of the top 30 neighbors and their corresponding edge weights in terms of $w_H(\cdot,\cdot)$ by weeks.

Week	Consecutive change ($s = i - 1$)		Accumulative change ($s = 1$)	
(ith)	$\texttt{diff}^{(i,s)}$	$\Delta w^{(i,s)}$	$\texttt{diff}^{(i,s)}$	$\Delta w^{(i,s)}$
2	3.45	0.19	3.45	0.19
3	3.57	0.19	5.90	0.22
4	3.22	0.18	7.57	0.22
5	0.81	0.05	8.10	0.21
6	0.95	0.06	8.38	0.21
7	1.19	0.06	8.72	0.20
8	2.07	0.09	9.35	0.19

are used to quantify the differences of top M neighbors and their corresponding linking weights for u at different epochs. The first measure is to count how many of the top M ($M = 30$ in the experiments) neighbors change between two epochs:

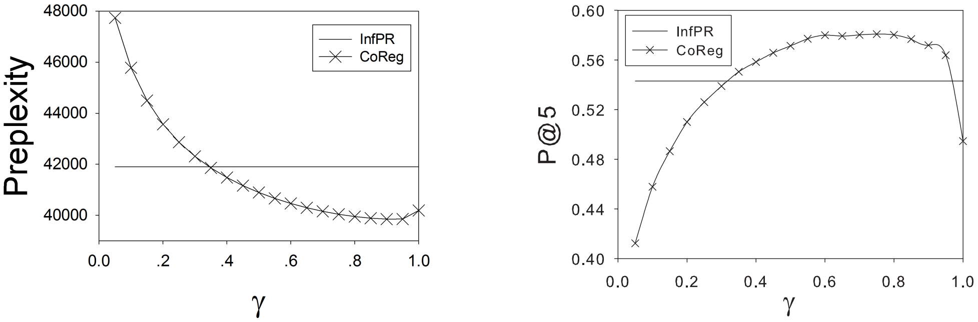
$$\begin{aligned}\texttt{diff}_A^{(i,s)}(u) &= |\mathcal{N}_{A,u}^{(i)}| - |\mathcal{N}_{A,u}^{(s)} \bigcap \mathcal{N}_{A,u}^{(i)}|,\\ \texttt{diff}_H^{(i,s)}(u) &= |\mathcal{N}_{H,u}^{(i)}| - |\mathcal{N}_{H,u}^{(s)} \bigcap \mathcal{N}_{H,u}^{(i)}|,\end{aligned}$$

If user v appears in the top M neighbors of user u at both the sth and ith epochs, we then use the following measure to compute the relative weight change between u and v.

$$\begin{aligned}\Delta w_A^{(i,s)}(u) &= \frac{1}{|\mathcal{N}_{A,u}^{(s)} \bigcap \mathcal{N}_{A,u}^{(i)}|} \sum_{v \in \mathcal{N}_{A,u}^{(s)} \bigcap \mathcal{N}_{A,u}^{(i)}} |w_A^{(s)}(u,v) - w_A^{(i)}(u,v)|,\\ \Delta w_H^{(i,s)}(u) &= \frac{1}{|\mathcal{N}_{H,u}^{(s)} \bigcap \mathcal{N}_{H,u}^{(i)}|} \sum_{v \in \mathcal{N}_{H,u}^{(s)} \bigcap \mathcal{N}_{H,u}^{(i)}} |w_H^{(s)}(u,v) - w_H^{(i)}(u,v)|.\end{aligned}$$

where $|w_H^{(s)}(u,v) - w_H^{(i)}(u,v)|$ is the absolute value of the difference between $w_H^{(s)}(u,v)$ and $w_H^{(i)}(u,v)$. In the above measures of $\texttt{diff}^{(i,s)}$ and $\Delta w^{(i,s)}$, for the ith epoch, we set s to 1 and $(i-1)$ respectively to compute the *accumulative change* (i.e., with respective to the 1st week) and *consecutive change* (i.e., with respective to the previous week). Having the values of these two measures, we further average them over all the users to see the overall change patterns.

We present the results in Table VIII and IX. It can be observed that the consecutive change of the first four weeks is more significant compared to the last four weeks. Also, after the first four weeks, the change seems to be relatively small and the network becomes more stable. These observations suggest that for a new user, her corresponding



(a) perplexity (#topic=50).

(b) precision@5.

Fig. 5. Parameter sensitivity of γ in CoReg. We take the optimal result of InfPR as a comparison.

link weights should be updated more regularly, e.g., on a weekly basis; while for a user with Twitter for a longer time, her weights can be updated less frequently, e.g., on a monthly basis. By doing so, it is possible to run our proposed method efficiently on a very large data set comprising of millions of users.

6.9. Parameters setting

In this section, we discuss how to set the parameters of our proposed method CoReg. Recall that there are in total three parameters in CoReg: α, β and γ. In our experiments, it generally works well when we have the same weights for both the in-link regularization and the out-link regularization, i.e., $\alpha = \beta$. With the constraint $\alpha+\beta+\gamma = 1$, we only have one parameter to tune, i.e., γ. Once γ is fixed, we have $\alpha = \beta = (1 - \gamma)/2$. To see how γ affects our method, we tune the values of γ from 0 to 1 with a step of 0.05. We present the results in Figure 5, and choose the optimal result from InfPR as a comparison. From this figure, we can see that CoReg performs consistently better than InfPR when the value of γ is beyond 0.4. Empirically, we notice that when we have more evidence (i.e., more data) from users themselves we should set a larger value for γ, and vice versa.

7. RELATED WORK

Our work is mainly related to the following topics:

Graph regularization: There is a very long history of the application of regularization techniques in machine learning [Neumaier and REV 1998], in which it was first used to find meaningful approximate solutions of ill-conditioned or singular systems. In statistics and machine learning, regularization techniques are mainly used to prevent over-fitting [Tibshirani 1994], e.g, L_2 regularization.

In recent years, with the emergence of various data with rich attributes (e.g., network links), regularization techniques are adopted in order to model these useful features. Specially, graph regularization methods have been proposed to utilize the graph structure underlying the data [Mei et al. 2008; Li et al. 2008; Ji et al. 2011; Zhou et al. 2004] and have been widely applied to various social media mining tasks, e.g., tag recommendation [Guan et al. 2009], object classification [Yin et al. 2009], and collaborative filtering [Ma et al. 2009a; Ma et al. 2011]. Apart from studies on undirected graph regularization, Zhou et al. [2005; 2005] proposed two methods to deal with directed graphs in the regularization framework. The main idea is to transform directed graphs to undirected ones, therefore the undirected regularization framework can be reused. Our work is partly inspired by the study in [Zhou et al. 2005]. While Zhou et al. [2005] aims to perform classification on data instances with directed links, we focus on a different task of learning user interests from online social networks taking into account various social connections.

Regularization methods have been applied on rating networks [Ma et al. 2011] and author networks [Mei et al. 2008]. Nevertheless, Twitter itself is more complex than those networks which makes our task more challenging. The most significant difference between our work and the previous research is the underlying assumptions made. Existing work is mainly built on the cluster assumption while ours is built on the novel link structure assumption, which is arguably more suitable for online social network analysis. Under the cluster assumption, node similarities are measured based on the existence of explicit links. However, under the link structure assumption, node similarities are evaluated based on the local link structures between two nodes. Our experimental results show that algorithms built on the link structure assumption is potentially more robust in complex and dynamic networks.

User interest modeling: Steyvers et al. [2004] proposed to represent user interests as topic distributions over topics in text . Yang et al. [2012] modeled user interests as a weighted term vector based on the tweets users posted, and then use a cosine function to calculate the user similarities. Yin et al. [2010] utilized tags that users have used to represent user interests. Some recent studies took temporal factors into consideration when modeling user interests [Yin et al. 2011; Ahmed et al. 2011]. In this article, following [Steyvers et al. 2004], we represent user interests as distributions over a set of fixed topics.

There have also been some research on analyzing and learning user interests from social networks [Han et al. 2012; Guo et al. 2009; Wang et al. 2011; Shi et al. 2009]. Our work is different from theirs in that we go beyond friend-link relationships and proposed a principled framework to leverage various types of social connections under the link structure assumption.

Topic mining in Twitter: Broadly speaking, there are two major ways to characterize and analyze Twitter topics, topic model based and hashtag based approaches. Zhao et al. [2011] proposed a Twitter-LDA which extracts topics from tweets concatenated for each user. Hong et al. [2012] proposed to extract meaningful topics by combining geography information. Another way to characterize topics is to utilize the mechanism of hashtagging. Kwak et al. [2010] used top ranked hashtags as trending topics in Twitter. Romero et al. [2011] performed a preliminary study of the adoption of hashtags. Based on that, Yang et al. [2012] further examined how dual role affects hashtag adoption. Lehmann et al. [2012] provided a temporal analysis of different types of hashtags. A way to combine content analysis and network regularization is to develop a joint topic model as in [Mei et al. 2008]. Our proposed method can work with any topic representations, including multinomial topic distributions or hashtag topic representations.

Other related work: “Homophily” is an important concept in online social network studies. Both traditional cluster assumption and our link structure assumption are formal ways to define and characterize “homophily” in social networks. Our method consists of two regularization factors, namely the in-link regularization factor and the out-link regularization factor. We propose in-link and out-link similarity functions for user similarity measurement. These functions are related to SimRank [Jeh and Widom 2002] which assumes that two objects are similar if they are related to similar objects. Nevertheless, there are a couple of notable differences between SimRank and our proposed similarity functions. First, SimRank can’t be used in sparse directed networks. Instead, our proposed similarity functions can be easily adapted to handle various relations (recall we have the relation weight function $w_R(\cdot, \cdot)$ in Eq. 2 and Eq. 4) in sparse directed networks. Second, SimRank is a recursive method for learning similarities which can be very time-consuming. When the structural context of one vertex (or object) changes, SimRank needs to be rerun. On the contrary, our method only stores the top M most similar neighbors for each user. Hence, when the structural context of a vertex changes, we only need to update the similarity scores of its top M neighbors. As have been discussed in Section 6.8, the similarity scores or weight scores of the links do not need to be updated very often. As such, our method is more efficient than SimRank.

8. CONCLUSIONS AND DISCUSSION

In this paper, we have proposed a novel framework to learn user interests from online social networks. In particular, we have introduced the link structure assumption for evaluating user similarities based on the local link structures instead of merely relying on explicit links, as is often the case in the previous studies. We have provided a principled solution to model implicit and directed social connections. Our proposed model outperforms a few competitive baselines on learning user interests over both multinomial topics and hashtag topics. Moreover, the experimental results on a range of tasks including retweet prediction and topical authority identification show that our model consistently performs the best.

Under the introduced link structure assumption, more complicated methods can be explored to define the user similarities such as counting the number of common triangles within the neighborhoods of two users in the relation graph. Nevertheless, such methods could potentially require larger time and memory complexity and therefore may not be scalable in large datasets. In this article, we have presented a relatively simple but effective way to compute the user similarities. We believe that our proposed framework will inspire more follow-up studies under the link structure assumption.

There are a few directions we would like to explore in the future. First, we transform directed graphs into undirected ones which may result in information loss due to the removal of link directions. One possible solution is to differentiate between explicit and implicit links and keep the direction information in explicit links. Second, we only consider the out-link and the in-link regularization factors under the link structure assumption. There could be other alternative methods to instantiate this assumption. We plan to investigate it further in the future.

ACKNOWLEDGEMENTS

We thank Jing Jiang for helping to improve the work. We thank the anonymous reviewers for the constructive comments. The work was paritially supported by the National Natural Science foundation of China under Grant No.61272340, 60933004, and Grant SKLSDE-2013KF. Yulan He was partially supported by the EPSRC grant EP/J020427/1. Xin Zhao was supported by Microsoft Research Asia Fellowship.

REFERENCES

AHMED, A., LOW, Y., ALY, M., JOSIFOVSKI, V., AND SMOLA, A. J. 2011. Scalable distributed inference of dynamic user interests for behavioral targeting. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '11. ACM, 114–122.

ANAGNOSTOPOULOS, A., KUMAR, R., AND MAHDIAN, M. 2008. Influence and correlation in social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '08. ACM, 7–15.

BAKSHY, E., HOFMAN, J. M., MASON, W. A., AND WATTS, D. J. 2011. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*. WSDM '11. ACM, 65–74.

BLEI, D. M., NG, A. Y., AND JORDAN, M. I. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*.

Feng, W. and Wang, J. 2013. Retweet or not?: personalized tweet re-ranking. In *Proceedings of the sixth ACM international conference on Web search and data mining*. WSDM '13. ACM, 577–586.

Griffiths, T. L. and Steyvers, M. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences 101*, 5228–5235.

Guan, Z., Bu, J., Mei, Q., Chen, C., and Wang, C. 2009. Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '09. ACM, 540–547.

Guo, L., Tan, E., Chen, S., Zhang, X., and Zhao, Y. E. 2009. Analyzing patterns of user content generation in online social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '09. ACM, 369–378.

Han, J., Sun, Y., Yan, X., and Yu, P. S. 2012. Mining knowledge from data: An information network analysis approach. In *Proceedings of International Conference on Data Engineering*. ICDE '12. IEEE.

Hong, L., Ahmed, A., Gurumurthy, S., Smola, A. J., and Tsioutsiouliklis, K. 2012. Discovering geographical topics in the twitter stream. In *Proceedings of the 21st international conference on World Wide Web*. WWW '12. ACM, 769–778.

Hong, L., Dan, O., and Davison, B. D. 2011. Predicting popular messages in twitter. In *Proceedings of the 20th international conference companion on World wide web*. WWW '11. ACM, 57–58.

Hong, L. and Davison, B. D. 2010. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*. SOMA '10. ACM, 80–88.

Hopcroft, J., Lou, T., and Tang, J. 2011. Who will follow you back?: reciprocal relationship prediction. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. CIKM '11. ACM, 1137–1146.

Jeh, G. and Widom, J. 2002. Simrank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '02. ACM, 538–543.

Ji, M., Yan, J., Gu, S., Han, J., He, X., Zhang, W. V., and Chen, Z. 2011. Learning search tasks in queries and web pages via graph regularization. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. SIGIR '11. ACM, 55–64.

Kwak, H., Lee, C., Park, H., and Moon, S. 2010. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*. WWW '10. ACM, 591–600.

Lehmann, J., Gonçalves, B., Ramasco, J. J., and Cattuto, C. 2012. Dynamical classes of collective attention in twitter. In *Proceedings of the 21st international conference on World Wide Web*. WWW '12. ACM, 251–260.

Li, X., Wang, Y.-Y., and Acero, A. 2008. Learning query intent from regularized click graphs. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '08. ACM, 339–346.

Ma, H., King, I., and Lyu, M. R. 2009a. Learning to recommend with social trust ensemble. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. SIGIR '09. ACM, 203–210.

Ma, H., Lyu, M. R., and King, I. 2009b. Learning to recommend with trust and distrust relationships. In *Proceedings of the third ACM conference on Recommender systems*. RecSys '09. ACM, 189–196.

Ma, H., Zhou, D., Liu, C., Lyu, M. R., and King, I. 2011. Recommender systems with social regularization. In *Proceedings of the fourth ACM international conference on Web search and data mining*. WSDM '11. ACM, New York, NY, USA, 287–296.

Manning, C. D., Raghavan, P., and Schtze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Mei, Q., Cai, D., Zhang, D., and Zhai, C. 2008. Topic modeling with network regularization. In *Proceedings of the 17th international conference on World Wide Web*. WWW '08. ACM, 101–110.

Neumaier, A. and REV, S. 1998. Solving Ill-Conditioned and Singular Linear Systems: A Tutorial on Regularization. *Siam Review 40*.

Romero, D. M., Meeder, B., and Kleinberg, J. 2011. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *Proceedings of the 20th international conference on World wide web*. WWW '11. ACM, 695–704.

Shi, X., Zhu, J., Cai, R., and Zhang, L. 2009. User grouping behavior in online forums. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '09. ACM, 777–786.

STEYVERS, M., SMYTH, P., ROSEN-ZVI, M., AND GRIFFITHS, T. 2004. Probabilistic author-topic models for information discovery. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '04. ACM, 306–315.

TANG, J., SUN, J., WANG, C., AND YANG, Z. 2009. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '09. ACM, 807–816.

TIBSHIRANI, R. 1994. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B 58*, 267–288.

WANG, C., RAINA, R., FONG, D., ZHOU, D., HAN, J., AND BADROS, G. 2011. Learning relevance from heterogeneous social network and its application in online targeting. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. SIGIR '11. ACM, 655–664.

WELCH, M. J., SCHONFELD, U., HE, D., AND CHO, J. 2011. Topical semantics of twitter links. In *Proceedings of the fourth ACM international conference on Web search and data mining*. WSDM '11. ACM, 327–336.

WENG, J., LIM, E.-P., JIANG, J., AND HE, Q. 2010. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*. WSDM '10. ACM, 261–270.

WU, S., HOFMAN, J. M., MASON, W. A., AND WATTS, D. J. 2011. Who says what to whom on twitter. In *Proceedings of the 20th international conference on World wide web*. WWW '11. ACM, 705–714.

YANG, L., SUN, T., ZHANG, M., AND MEI, Q. 2012. We know what @you #tag: does the dual role affect hashtag adoption? In *Proceedings of the 21st international conference on World Wide Web*. WWW '12. ACM, 261–270.

YIN, D., HONG, L., XUE, Z., AND DAVISON, B. D. 2011. Temporal dynamics of user interests in tagging systems. In *Proceedings of 25th AAAI Conference on Artificial Intelligence*. AAA '11.

YIN, D., XUE, Z., HONG, L., AND DAVISON, B. D. 2010. A probabilistic model for personalized tag prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '10. ACM, 959–968.

YIN, Z., LI, R., MEI, Q., AND HAN, J. 2009. Exploring social tagging graph for web object classification. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. KDD '09. ACM, 957–966.

ZHAO, W. X., JIANG, J., WENG, J., HE, J., LIM, E.-P., YAN, H., AND LI, X. 2011. Comparing twitter and traditional media using topic models. In *Proceedings of the 33rd European conference on Advances in information retrieval*. ECIR'11. Springer-Verlag, 338–349.

ZHOU, D., BOUSQUET, O., LAL, T. N., WESTON, J., AND SCHÖLKOPF, B. 2004. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*. NIPS '04. MIT Press.

ZHOU, D., HUANG, J., AND SCHÖLKOPF, B. 2005. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd international conference on Machine learning*. ICML '05. ACM, 1036–1043.

ZHOU, D., SCHÖLKOPF, B., AND HOFMANN, T. 2005. Semi-supervised learning on directed graphs. In *Advances in Neural Information Processing Systems 17*. MIT Press, 1633–1640.