Online Social Spammer Detection

Xia Hu, Jiliang Tang, Huan Liu
Computer Science and Engineering, Arizona State University, USA
{xiahu, jiliang.tang, huan.liu}@asu.edu

Abstract

The explosive use of social media also makes it a popular platform for malicious users, known as social spammers, to overwhelm normal users with unwanted content. One effective way for social spammer detection is to build a classifier based on content and social network information. However, social spammers are sophisticated and adaptable to game the system with fast evolving content and network patterns. First, social spammers continually change their spamming content patterns to avoid being detected. Second, reflexive reciprocity makes it easier for social spammers to establish social influence and pretend to be normal users by quickly accumulating a large number of "human" friends. It is challenging for existing anti-spamming systems based on batch-mode learning to quickly respond to newly emerging patterns for effective social spammer detection. In this paper, we present a general optimization framework to collectively use content and network information for social spammer detection, and provide the solution for efficient online processing. Experimental results on Twitter datasets confirm the effectiveness and efficiency of the proposed framework.

Introduction

Social media services, like Facebook and Twitter, are increasingly used in various scenarios such as marketing, journalism and public relations. While social media services have emerged as important platforms for information dissemination and communication, it has also become infamous for spammers who overwhelm other users with unwanted content. The (fake) accounts, known as social spammers (Webb *et al.* 2008; Lee *et al.* 2010), are a special type of spammers who coordinate among themselves to launch various attacks such as spreading ads to generate sales, disseminating pornography, viruses, phishing, befriending victims and then surreptitiously grabbing their personal information (Bilge *et al.* 2009), or simply sabotaging a system's reputation (Lee *et al.* 2010). The problem of social spamming is a serious issue prevalent in social media sites. Characterizing and detecting social spammers can significantly improve the quality of user experience, and promote the healthy use and development of a social networking system.

Following spammer detection in traditional platforms like Email and the Web (Chen *et al.* 2012), some efforts have been devoted to detect spammers in various social networking sites, including Twitter (Lee *et al.* 2010), Renren (Yang *et al.* 2011), Blogosphere (Lin *et al.* 2007), etc. Existing methods can generally be divided into two categories. First category is to employ content analysis for detecting spammers in social media. Profile-based features (Lee *et al.* 2010) such as content and posting patterns are extracted to build an effective supervised learning model, and the model is applied on unseen data to filter social spammers. Another category of methods is to detect spammers via social network analysis (Ghosh *et al.* 2012). A widely used assumption in the methods is that spammers cannot establish an arbitrarily large number of social trust relations with legitimate users. The users with relatively low social influence or social status in the network will be determined as spammers.

Traditional spammer detection methods become less effective due to the fast evolution of social spammers. First, social spammers show dynamic content patterns in social media. Spammers' content information changes too fast to be detected by a static anti-spamming system based on offline modeling (Zhu *et al.* 2012). Spammers continue to change their spamming strategies and pretend to be normal users to fool the system. A built system may become less effective when the spammers create many new, evasive accounts. Second, many social media sites like Twitter have become a target of link farming (Ghosh *et al.* 2012). The reflexive reciprocity (Weng *et al.* 2010; Hu *et al.* 2013b) indicates that many users simply follow back when they are followed by someone for the sake of courtesy. It is easier for spammers to acquire a large number of follower links in social media. Thus, with the perceived social influence, they can avoid being detected by network-based methods. Similar results targeting other platforms such as Renren (Yang *et al.* 2011) have been reported in literature as well. Existing systems rely on building a new model to capture newly emerging content-based and network-based patterns of social spammers. Given the rapidly evolving nature, it is necessary to have a framework that efficiently reflects the effect of newly emerging data.

One efficient approach to incrementally update existing model in large-scale data analysis is *online learning*. While online learning has been studied for years and shown its effectiveness in many applications such as image and video processing (Mairal *et al.* 2009) and human computer in-

teraction (Madani *et al.* 2009), it has not been applied in social spammer detection. In this paper, we study how to capture the fast evolving nature of social spammers using online learning. In particular, we investigate: (1) how do we model the content and network information in a unified framework for effective social spammer detection?; and (2) how do we update the built model to efficiently incorporate newly emerging data objects? Our solutions to these two questions result in a new framework for Online Social Spammer Detection (*OSSD*). The proposed framework is a formulation based on directed Laplacian constrained matrix factorization, and is used to incorporate refined social network information into content modeling. Then we incrementally update the factors appropriately to reflect the rapidly evolving nature of the social spammers. The main contributions of this paper are outlined as follows:

- Formally define the problem of online social spammer detection with content and network information;
- Introduce a unified framework that considers both of the content and network information for effective social spammer detection;
- Propose a novel scheme to incrementally update the built model for social spammer detection; and
- Empirically evaluate the proposed *OSSD* framework on real-world Twitter datasets.

The remainder of this paper is organized as follows. In the second section, we formally define the problem of online social spammer detection. We then introduce a general framework for social spammer detection with content and network information. In the fourth section, we propose a novel online learning scheme for the social spammer detection framework. In the experiment section, we report empirical results on real-world Twitter datasets to evaluate the effectiveness and efficiency of the proposed method. Finally, we conclude the paper and present the future work.

Problem Statement

In this section, we first introduce the notations used in the paper and then formally define the problem we study.

Notation: Scalars are denoted by lowercase letters, vectors by boldface lowercase letters, and matrices by boldface uppercase letters. Let $\|\mathbf{A}\|$ denote the Euclidean norm, and $\|\mathbf{A}\|_F$ the Frobenius norm of the matrix $\mathbf{A}$, i.e., $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\mathbf{A}_{ij}^2}$. Let $\mathbf{A}^T$ denote the transpose of $\mathbf{A}$.

Let $[\mathbf{X}, \mathcal{G}, \mathbf{Y}]$ be a target social media user set with content information of social media posts $\mathbf{X}$, social network information $\mathcal{G}$, and identity label matrix $\mathbf{Y}$. We use $\mathbf{X} \in \mathbb{R}^{n\times m}$ to denote content information, i.e., messages posted by the users, where n is the number of textual features and m is the number of users. We use $\mathcal{G} = (V, E)$ to denote the social network, where nodes u and v in V represent social media users, and each directed edge $[u, v]$ in E represents a following relation from u to v. We do not have self links in the graph, i.e., $u \neq v$. We use $\mathbf{Y} \in \mathbb{R}^{m\times c}$ to denote the identity label matrix, where c is the number of identity labels. Following literature on spammer detection (Benevenuto *et al.* 2010; Lee *et al.* 2010), we focus on classifying users as spammers or normal users, i.e., $c = 2$. It is straightforward to extend this setting to a multi-class classification task.

With the given notations, we formally define the problem of online social spammer detection as follows:

Given k users with their content information $\mathbf{X}^k$, social network information $\mathbf{G}^k$, and identity label information $\mathbf{Y}^k$, we learn a factorization model $\mathbf{V}^k$ and $\mathbf{U}^k$ which could be used to learn a classifier $\mathbf{W}^k$ to automatically assign identity labels for unknown users (i.e., test data) as spammers or normal users. Given one more user, our goal is to efficiently update the built model $\mathbf{V}^{k+1}$, $\mathbf{U}^{k+1}$ and $\mathbf{W}^{k+1}$ for social spammer detection based on $k+1$ users with their content information $\mathbf{X}^{k+1}$, social network information $\mathbf{G}^{k+1}$, and identity label information $\mathbf{Y}^{k+1}$.

Social Spammer Detection

In this section, we propose a general framework for social spammer detection. We first discuss the modeling of content and social network information separately, and then introduce a unified framework to integrate both information.

To use content information, one way is to learn a supervised model, and apply the learned model for spammer detection. Due to the unstructured and noisy content information in social media, this method yields two problems to be directly applied to our task. First, text representation models, like n-gram model, often lead to a high-dimensional feature space because of the large size of data and vocabulary (Hu *et al.* 2009). Second, In addition to the short form of texts, abbreviations and acronyms are widely used in social media, thus making the data representation very sparse.

To tackle the problems, instead of learning word-level knowledge, we propose to model the content information from topic-level. Motivated by topic modeling literature (Blei *et al.* 2003), a user's posts usually focus on a few topics, resulting in $\mathbf{X}$ very sparse and low-rank. The proposed method is built on a non-negative matrix factorization model (NMF) (Lee and Seung 1999), which seeks a more compact but accurate low-rank representation of the users by solving the following optimization problem:

$$\min_{\mathbf{U},\mathbf{H}\geq 0} \quad \|\mathbf{X} - \mathbf{U}\mathbf{H}\|_F^2, \tag{1}$$

where $\mathbf{X}$ is the content matrix, $\mathbf{U} \in \mathbb{R}^{n\times r}$ is a mixing matrix and $\mathbf{H} \in \mathbb{R}^{r\times m}$ with $r \ll n$ is an encoding matrix that indicates a low-rank user representation in a topic space.

Previous studies have shown that social network information is helpful in many applications such as sentiment analysis (Tan *et al.* 2011), trust prediction (Tang *et al.* 2013) and community deviation detection (Chen *et al.* 2010). A widely used assumption is that representations of two nodes are close when they are connected with each other in the network (Chung 1997; Zhu *et al.* 2012). This assumption does not hold in social media. Some social media services such as microblogging have directed following relations between users. In addition, it is practical for social spammers to quickly attract a large number of followers to fool the system. Thus it is not suitable to directly apply the existing methods to the problem we study.

Following the way used in (Hu *et al.* 2013b) to model social network information, we employ a variant of directed graph Laplacian to model social network information. Given the social network information $\mathcal{G}$ and the identity label matrix $\mathbf{Y}$, there are four kinds of following relations:

[spammer, spammer], [normal, normal],

[normal, spammer], [spammer, normal].

The fourth relation that can be easily faked by spammers. We exclude the fourth relation and only make use of the first three relations. Thus, the adjacency matrix $\mathbf{G} \in \mathbb{R}^{m \times m}$ is defined as

$$\mathbf{G}(u,v) = \begin{cases} 1 & \text{if [u, v] is among the first three relations} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where u and v are users, and $[u, v]$ is a directed edge in the graph $\mathcal{G}$.

The in-degree and out-degree of node u is defined as $\mathbf{d}_u^{in} = \sum_{[v,u]} \mathbf{G}(v,u)$ and $\mathbf{d}_u^{out} = \sum_{[u,v]} \mathbf{G}(u,v)$. Let $\mathbf{P}$ be the transition probability matrix of random walk in a given graph with $\mathbf{P}(u,v) = \mathbf{G}(u,v)/\mathbf{d}_u^{out}$ (Zhou *et al.* 2005). The random walk has a stationary distribution $\boldsymbol{\pi}$, which satisfy $\sum_{u \in V} \boldsymbol{\pi}(u) = 1$, $\boldsymbol{\pi}(v) = \sum_{[u,v]} \boldsymbol{\pi}(u)\mathbf{P}(u,v)$ (Chung 2005; Zhou *et al.* 2005), and $\boldsymbol{\pi}(u) > 0$ for all $u \in V$.

To model the network information, the basic idea is to make the latent representations of two users as close as possible if there exists a following relation between them. It can be mathematically formulated as minimizing

$$\begin{aligned} \mathcal{R} = & \frac{1}{2} \sum_{[u,v] \in E} \boldsymbol{\pi}(u)\mathbf{P}(u,v) \|\mathbf{H}_u - \mathbf{H}_v\|^2 \\ = & \ tr(\mathbf{H}(\boldsymbol{\Pi} - \frac{\boldsymbol{\Pi}\mathbf{P} + \mathbf{P}^T\boldsymbol{\Pi}}{2})\mathbf{H}^T) \\ = & \ tr(\mathbf{H}\mathcal{L}\mathbf{H}^T), \end{aligned} \tag{3}$$

where $\mathbf{H}_u$ denotes the low-rank representation of user u, $\mathbf{H}_v$ the low-rank representation of user v, and $\boldsymbol{\Pi}$ denotes a diagonal matrix with $\boldsymbol{\Pi}(u,u) = \boldsymbol{\pi}(u)$. The induction of Eq. (3) is straightforward and can be also found in previous work (Chung 2005; Zhou *et al.* 2005). This loss function will incur a penalty if two users have different low-rank representations when they have a directed relation in the graph.

With the NMF model, we project the original content information into a latent topic space. By adding the network information discussed in Eq. (3) as a regularization, our proposed framework can be mathematically formulated as solving the following optimization problem:

$$\min_{\mathbf{H},\mathbf{U} \geq 0} \ \mathcal{J} = \|\mathbf{X} - \mathbf{U}\mathbf{H}\|_F^2 + \alpha\mathcal{R}, \tag{4}$$

where α is the regularization parameter to control the effects of social network information to the learned model.

The objective function defined in Eq. (4) is convex of $\mathbf{U}$ and $\mathbf{H}$ separately. Following the multiplicative and alternating updating rules introduced in (Seung and Lee 2001), we optimize the objective with respect to one variable, while fixing the other. Since $\mathcal{L}$ may take any signs, we decompose it as $\mathcal{L} = \mathcal{L}^+ - \mathcal{L}^-$. The updating rules for the variables are:

$$\mathbf{U}(i,j) \leftarrow \mathbf{U}(i,j)\sqrt{\frac{[\mathbf{X}\mathbf{H}^T](i,j)}{[\mathbf{U}\mathbf{H}\mathbf{H}^T](i,j)}}, \tag{5}$$

$$\mathbf{H}(i,j) \leftarrow \mathbf{H}(i,j)\sqrt{\frac{[\mathbf{U}^T\mathbf{X} + \alpha\mathbf{H}\mathcal{L}^-](i,j)}{[\mathbf{U}^T\mathbf{U}\mathbf{H} + \alpha\mathbf{H}\mathcal{L}^+](i,j)}}. \tag{6}$$

The correctness and convergence of the updating rules can be proven with the standard auxiliary function approach (Seung and Lee 2001; Gu *et al.* 2010). Once obtaining the low-rank user representation $\mathbf{H}$, a supervised model can be trained based on the new latent topic space. We employ the widely used Least Squares (Lawson and Hanson 1995), which has a closed-form solution: $\mathbf{W} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{Y}$.

Online Social Spammer Detection

Online learning is an efficient approach to incrementally update existing model in large-scale data processing. While online learning has been widely used in various applications such as computer vision (Bucak and Gunsel 2009; Mairal *et al.* 2010), speech recognition (Wang *et al.* 2013) and bioinformatics (Yang *et al.* 2010), the application to spammer detection is a very new effort. In this section, we will discuss the use of online learning scheme, instead of batch-mode learning, to update the built social spammer detection model.

We have introduced a general social spammer detection model in last section. Given a model built on k users, the aim of the proposed method *OSSD* is to update factor matrices $\mathbf{U}$ and $\mathbf{H}$ by adding the $(k+1)th$ user without much computational effort. Following the formulation in Eq. (4), the objective function for $k+1$ users is defined as

$$\min_{\mathbf{U}^{k+1},\mathbf{H}^{k+1} \geq 0} \mathcal{J}^{k+1} = \|\mathbf{X}^{k+1} - \mathbf{U}^{k+1}\mathbf{H}^{k+1}\|_F^2 + \alpha\mathcal{R}^{k+1}, \tag{7}$$

where $\mathbf{X}^{k+1}$ represents the content matrix of $k+1$ users, $\mathbf{U}^{k+1}$ and $\mathbf{H}^{k+1}$ denote the factor matrices to be updated, and $\mathcal{R}^{k+1}$ indicates the objective function of graph Laplacian. This optimization problem can be solved with the batch-mode learning updating rules given by Eqs. (5) and (6). However, due to its high computational cost, an online learning updating scheme is needed.

Columns of mixing matrix $\mathbf{U}$ can be considered as the building blocks of the data, and each entity of $\mathbf{H}$ determines how the building blocks involved in the corresponding observation in $\mathbf{X}$ (Hoyer 2004). As the number of data objects increases, effects of each object on the representation decrease. Since the new data objects would not be able to significantly change the mixing matrix $\mathbf{U}$, it is not necessary to update the part of original encoding matrix $\mathbf{H}$ which corresponds to old objects. Thus, besides updating the mixing matrix $\mathbf{U}$, it is adequate to only update the last column of $\mathbf{H}_{k+1}$ by assuming the first k columns of $\mathbf{H}_{k+1}$ would be approximately equal to $\mathbf{H}_k$. The objective function in

Eq. (7) can be reformulated as:

$$\begin{aligned}
\mathcal{J}^{k+1} &= \|\mathbf{X}^{k+1} - \mathbf{U}^{k+1}\mathbf{H}^{k+1}\|_F^2 \\
&+\alpha \sum_{i=1}^{k+1}\sum_{j=1}^{k+1} \boldsymbol{\pi}(i)\mathbf{P}(i,j)\|\mathbf{H}_i - \mathbf{H}_j\|^2 \\
&= \sum_{i=1}^{n}\sum_{j=1}^{k+1}(\mathbf{X}^{k+1}(i,j) - (\mathbf{U}^{k+1}\mathbf{H}^{k+1})(i,j))^2 \\
&+\alpha \sum_{i=1}^{k+1}\sum_{j=1}^{k+1} \boldsymbol{\pi}(i)\mathbf{P}(i,j)\|\mathbf{H}_i - \mathbf{H}_j\|^2 \\
&\approx \sum_{i=1}^{n}\sum_{j=1}^{k}(\mathbf{X}^{k}(i,j) - (\mathbf{U}^{k+1}\mathbf{H}^{k})(i,j))^2 \\
&+ \sum_{i=1}^{n}(\mathbf{X}^{k+1}(i,k+1) - (\mathbf{U}^{k+1}\mathbf{H}^{k+1})(i,k+1))^2 \\
&+\alpha \sum_{i=1}^{k}\sum_{j=1}^{k} \boldsymbol{\pi}(i)\mathbf{P}(i,j)\|\mathbf{H}_i - \mathbf{H}_j\|^2 \\
&+2\alpha \sum_{j=1}^{k} \boldsymbol{\pi}(k+1)\mathbf{P}(k+1,j)\|\mathbf{H}_{k+1} - \mathbf{H}_j\|^2,
\end{aligned}$$

and it can be further reformulated as:

$$\begin{aligned}
\mathcal{J}^{k+1} \approx\ & 2\alpha \sum_{j=1}^{k} \boldsymbol{\pi}(k+1)\mathbf{P}(k+1,j)\|\mathbf{H}_{k+1} - \mathbf{H}_j\|^2 \\
&+ \sum_{i=1}^{n}(\mathbf{X}^{k+1}(i,k+1) - (\mathbf{U}^{k+1}\mathbf{H}^{k+1})(i,k+1))^2 + \mathcal{J}^k,
\end{aligned}$$

where $\mathcal{J}^k$ is the objective function for k users defined in Eq. (4). Following the updating rules introduced in (Seung and Lee 2001), gradient descent optimization that yields *OSSD* is performed. When a new data object arrives, the updating rules for the variables are:

$$\mathbf{H}^{k+1}(i,k+1) \leftarrow \mathbf{H}^{k+1}(i,k+1)\sqrt{\frac{[\mathbf{A}](i,1)}{[\mathbf{B}](i,1)}},$$

$$\mathbf{U}^{k+1}(i,j) \leftarrow \mathbf{U}^{k+1}(i,j)\sqrt{\frac{[\mathbf{X}^k\mathbf{H}^{k^T} + \mathbf{C}](i,j)}{[\mathbf{U}^{k+1}\mathbf{H}^k\mathbf{H}^{k^T} + \mathbf{D}](i,j)}},$$

where

$$\begin{aligned}
\mathbf{A} &= \mathbf{U}^{k+1^T}\mathbf{X}^{k+1}(*,k+1), \\
\mathbf{B} &= \mathbf{U}^{k+1^T}\mathbf{U}^{k+1}\mathbf{H}^{k+1}(*,k+1), \\
\mathbf{C} &= \mathbf{X}^{k+1}(*,k+1)\mathbf{H}^{k+1^T}(k+1,*), \\
\mathbf{D} &= \mathbf{U}^{k+1}\mathbf{H}^{k+1}(*,k+1)\mathbf{H}^{k+1^T}(k+1,*).
\end{aligned}$$

We present the algorithm of online social spammer detection in Algorithm 1. In the algorithm, we conduct initialization for the two matrices to be inferred in line 1. I is the number of maximum iterations. The two matrices are firstly learned with the method we discussed in last section, and then updated with the updating rules until convergence or reaching the number of maximum iterations from line 3 to 8. The classifier $\mathbf{W}$ is learned in line 9.

Algorithm 1: *Online Social Spammer Detection*

Input: $\{\mathbf{X}, \mathbf{Y}, \mathbf{G}, \alpha, I\}$
Output: $\mathbf{U}, \mathbf{H}, \mathbf{W}$

1 Initialize $\mathbf{U}, \mathbf{H} \geq 0$
2 Learning $\mathbf{U}^k, \mathbf{H}^k \geq 0$
3 **while** *Not convergent and iter* $\leq I$ **do**
4 Update $\mathbf{H}^{k+1}(i,k+1) \leftarrow$
5 $\mathbf{H}^{k+1}(i,k+1)\sqrt{\frac{[\mathbf{U}^{k+1^T}\mathbf{X}^{k+1}(*,k+1)](i,1)}{[\mathbf{U}^{k+1^T}\mathbf{U}^{k+1}\mathbf{H}^{k+1}(*,k+1)](i,1)}}$
6 Update $\mathbf{U}^{k+1}(i,j) \leftarrow$
7 $\mathbf{U}^{k+1}(i,j)\sqrt{\frac{[\mathbf{X}^k\mathbf{H}^{k^T}+\mathbf{C}](i,j)}{[\mathbf{U}^{k+1}\mathbf{H}^k\mathbf{H}^{k^T}+\mathbf{D}](i,j)}}$
8 $iter = iter + 1$
9 $\mathbf{W} = (\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{H}\mathbf{Y}$
10 **return** $\mathbf{W}$

The updating rule in Eq. (8) is helpful in reducing the computational cost. Since $\mathbf{X}^k$ and $\mathbf{H}^k$ do not change through the learning process, instead of storing $\mathbf{X}^k$ and $\mathbf{H}^k$, there are two benefits to store results of the matrix multiplications $\mathbf{X}^k\mathbf{H}^{k^T}$ and $\mathbf{H}^k\mathbf{H}^{k^T}$. First, the dimensions of the multiplications remain the same, thus the required storage memory will be the same regardless the sizes of $\mathbf{X}^k$ and $\mathbf{H}^k$. Second, the number of matrix multiplication is the main reason of the computational complexity of traditional NMF, and it will be significantly reduced through the process with the proposed online learning scheme.

In summary, we only update columns of the encoding matrix that correspond to the new data objects in Eq. (8), and the updating rule in Eq. (8) helps in reducing the computational cost. Thus, the proposed online learning scheme is more efficient. Comparing with traditional NMF with time complexity $O(nmr^2)$, the overall time complexity of the proposed *OSSD* is $O(nr^2)$, which is independent of the number of samples m.

Experiments

In this section, we conduct extensive experiments to evaluate the effectiveness and efficiency of the proposed framework *OSSD*. Through the experiments, we aim to answer the following two questions:

1. How effective is the proposed framework compared with other methods of social spammer detection?
2. How efficient is the proposed online learning framework compared with other methods for modeling?

Datasets

We now introduce two real-world Twitter datasets.

TAMU Social Honeypots Dataset (TwitterT):[1] This dataset was originally collected from December 30, 2009

[1] http://infolab.tamu.edu/data/

Table 1: Statistics of the Datasets

	TwitterT	*TwitterS*
# of Spammers	12,035	2,049
# of Legitimate Users	10,912	11,085
# of Tweets	2,530,516	380,799
Min Degree of Users	3	3
Max Degree of Users	1,312	1,025

to August 2, 2010 on Twitter and introduced in (Lee *et al.* 2011). It consists of Twitter users with identity labels: spammers and legitimate users. The dataset contains users, their number of followers and tweets. We filtered the non-English tweets and users with less than two tweets or two social connections. The corpus used in our study consists of 12,035 spammers and 10,912 legitimate users.

Twitter Suspended Spammers Dataset (TwitterS): Following the data crawling process used in (Yang *et al.* 2011; Zhu *et al.* 2012), we crawled this Twitter dataset from July to September 2012 via the Twitter Search API. The users that were suspended by Twitter during this period are considered as the gold standard (Thomas *et al.* 2011) of spammers in the experiment. We then randomly sampled the legitimate users from a publicly available Twitter dataset provided by TREC 2011.[2] According to literature (Lee *et al.* 2010) of spammer detection, the two classes are imbalanced, i.e., the number of legitimate users is much greater than that of spammers in the dataset. We filtered the non-English tweets and users with less than two tweets or two social connections.

The statistics of the two datasets are presented in Table 1.

Experimental Setup

We conduct two sets of experiments for evaluation. In the first set of experiments, we follow standard experiment settings used in (Benevenuto *et al.* 2010; Zhu *et al.* 2012) to evaluate the performance of spammer detection methods. In particular, we apply different methods on the Twitter datasets, and F_1-measure is used as the performance metric. In the second set of experiments, we compare efficiency of the proposed online learning scheme and batch-mode learning algorithms. Execution time is used as the performance metric. A standard procedure for data preprocessing is used in our experiments. We remove stop-words and perform stemming for all the tweets. The unigram model is employed to construct the feature space, tf-idf is used as the feature weight. One positive parameters α is involved in the experiments. α is to control the contribution of social network information. As a common practice, all the parameters can be tuned via cross-validation with validation data. In the experiments, we empirically set $\alpha = 0.1$ for experiments.

Effectiveness Evaluation

To answer the first question asked in the beginning of this section, we compare the proposed framework with following baseline methods for social spammer detection.

[2]http://trec.nist.gov/data/tweets/

- *LS_Content*: the Least Squares (Lawson and Hanson 1995) is a widely used classification method in many applications. We apply the Least Squares on the content matrix $\mathbf{X}$ for spammer detection.
- *LS_Net*: we apply the Least Squares on the adjacency matrix $\mathbf{G}$ of the social network for spammer detection.
- *MLSI*: this method considers both network and content information for spammer detection. Multi-label informed latent semantic indexing (Yu *et al.* 2005; Zhu *et al.* 2012) is used to model the content information, and undirected graph Laplacian (Chung 1997) is used to incorporate the network information.
- *BSSD*: this is a variant of our proposed method. Instead of online learning, we use batch-mode learning to build the model based on the training data at one time.
- *OSSD*: our proposed online learning method.

Among the five methods, the first four are based on batch-mode learning and the last one is designed using online learning. The experimental results of the methods are summarized in Table 2 and 3. In the experiments, five-fold cross-validation is used for all the methods. To study the effects brought by different sizes of training data, we varies the training data from 10% to 100%. In particular, for each round of the experiment, 20% of the dataset is held for testing and 10% to 100% of the original training data is sampled for training. For example, "50%" indicates that we use 50% of the 80%, thus using 40% of the whole dataset for training. For *OSSD*, the online learning updates a basic model that is built based on 50% of the training data in each round. In the table, "gain" represents the percentage improvement of the methods in comparison with the first baseline method *LS_Content*. In the experiment, each result denotes an average of 10 test runs. By comparing the results of different methods on the two datasets, we draw the following observations:

(1) From the results in the tables, we can observe that our proposed methods *BSSD* and *OSSD* consistently outperform other baseline methods on both datasets with different sizes of training data. Our spammer detection methods achieves better results than the state-of-the-art method *MLSI* on both datasets. We apply two-sample one-tail t-tests to compare *BSSD* and *OSSD* with the three baseline methods. The experiment results demonstrate that the proposed models perform significantly better (with significance level $\alpha = 0.01$) than the three baseline methods.

(2) The last three methods achieve better results than the first two methods that are based on only one type of information. The network-based method *LS_Net* achieves the worst performance among all the methods. This demonstrates that the integration of both content and network information is helpful for effective social spammer detection.

(3) The last two methods, *OSSD* and *BSSD*, achieve comparably good performance on both datasets with different sizes of training data. This shows that, comparing with batch-mode learning method, our proposed online learning scheme does not bring in any negative effects to the accuracy of social spammer detection.

Table 2: Social Spammer Detection Results on TwitterT Dataset

	10% (gain)	25% (gain)	50% (gain)	100% (gain)
LS_Content	0.803 (N.A.)	0.829 (N.A.)	0.838 (N.A.)	0.854 (N.A.)
LS_Net	0.625 (-22.17%)	0.640 (-22.80%)	0.609 (-27.33%)	0.611 (-28.45%)
MLSI	0.865 (+7.72%)	0.882 (+6.39%)	0.873 (+4.18%)	0.896 (+4.92%)
BSSD	0.878 (+9.34%)	0.901 (+8.69%)	0.909 (+8.47%)	0.921 (+7.85%)
OSSD	0.870 (+8.34%)	0.905 (+9.17%)	0.907 (+8.23%)	0.918 (+7.49%)

Table 3: Social Spammer Detection Results on TwitterS Dataset

	10% (gain)	25% (gain)	50% (gain)	100% (gain)
LS_Content	0.775 (N.A.)	0.801 (N.A.)	0.811 (N.A.)	0.829 (N.A.)
LS_Net	0.603 (-22.19%)	0.610 (-23.85%)	0.612 (-24.54%)	0.597 (-27.99%)
MLSI	0.838 (+8.13%)	0.851 (+6.24%)	0.859 (+5.92%)	0.879 (+6.03%)
BSSD	0.849 (+9.55%)	0.863 (+7.74%)	0.871 (+7.40%)	0.908 (+9.53%)
OSSD	0.843 (+8.77%)	0.865 (+7.99%)	0.873 (+7.64%)	0.906 (+9.29%)

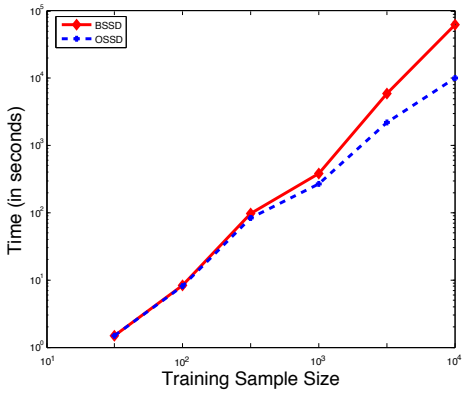

Figure 1: Efficiency Performance on TwitterT

In summary, the superior performance of our proposed method answers the first question that, compared with other methods, *OSSD* is effective in spammer detection. In addition, the proposed online learning scheme can achieve comparable performance with batch-mode learning methods. Next, we evaluate efficiency of the proposed method.

Efficiency Evaluation

To answer the second question asked in the beginning of this section, we compare the efficiency of batch-mode learning method *BSSD* with online learning based method *OSSD*. The experiments are run on a single-CPU, eight-core 3.40Ghz machine. Experimental results of the two methods on TwitterT dataset are plotted in Figure 1. in the figure, x axis represents the training sample size and y axis indicates the execution time in seconds of the methods. The red curve shows the performance of *BSSD* and the blue dotted curve depicts the performance of *OSSD*.

From the figure, we observe that the online version of our algorithm *OSSD* needs less running time than the batch-mode learning algorithm *BSSD*. This demonstrates that, our proposed online learning based method is more efficient than the batch-mode learning method. In many situations, especially when the training sample size is large, the differences in performance are significant between online learning and batch-mode learning method. Similar results have been observed on the TwitterS dataset; we omit the results owing to lack of space. In summary, the observations answer the second question that, comparing with other methods, online learning is efficient for social spammer detection.

Conclusion and Future Work

Social spammers are sophisticated and adaptable to game the system by continually change their content and network patterns. To handle fast evolving social spammers, we proposed to use online learning to efficiently reflect the newly emerging patterns. In this paper, we develop a general social spammer detection framework with both content and network information, and provide its online learning updating rules. In particular, we use directed graph Laplacian to model social network information, which is further integrated into a matrix factorization framework for content information modeling. By investigating its online updating scheme, we provide an efficient way for social spammer detection. Experimental results show that our proposed method is effective and efficient comparing with other social spammer detection methods.

This work suggests some interesting directions for future work. Besides network and content information, it would be interesting to study more user profile patterns (Hu *et al.* 2013a) such as gender, sentiment, location and political orientation of users for social spammer detection. In addition, the contribution of each data sample to the objective function is considered equal in our work. We can further investigate measures of importance of data objects to improve performance of the proposed online learning algorithm.

Acknowledgments

We truly thank the anonymous reviewers for their pertinent comments. This work is, in part, supported by ONR (N000141410095) and ARO (#025071).

References

F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting spammers on twitter. In *Proceedings of CEAS*, 2010.

L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda. All your contacts are belong to us: automated identity theft attacks on social networks. In *Proceedings of WWW*, 2009.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

Serhat S Bucak and Bilge Gunsel. Incremental subspace learning via non-negative matrix factorization. *Pattern Recognition*, 42(5):788–797, 2009.

Zhengzhang Chen, Kevin A Wilson, Ye Jin, William Hendrix, and Nagiza F Samatova. Detecting and tracking community dynamics in evolutionary networks. In *ICDMW*, pages 318–327, 2010.

Zhengzhang Chen, William Hendrix, and Nagiza F Samatova. Community-based anomaly detection in evolutionary networks. *JIIS*, 2012.

F.R.K. Chung. *Spectral graph theory*. Number 92. Amer Mathematical Society, 1997.

F. Chung. Laplacians and the cheeger inequality for directed graphs. *Annals of Combinatorics*, 9(1):1–19, 2005.

S. Ghosh, B. Viswanath, F. Kooti, N.K. Sharma, G. Korlam, F. Benevenuto, N. Ganguly, and K.P. Gummadi. Understanding and combating link farming in the twitter social network. In *Proceedings of WWW*, 2012.

Quanquan Gu, Jie Zhou, and Chris HQ Ding. Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In *SDM*, pages 199–210, 2010.

Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *The Journal of Machine Learning Research*, 5:1457–1469, 2004.

Xia Hu, Nan Sun, Chao Zhang, and Tat-Seng Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of CIKM*, 2009.

Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. Unsupervised sentiment analysis with emotional signals. In *Proceedings of WWW*, 2013.

Xia Hu, Jiliang Tang, Yanchao Zhang, and Huan Liu. Social spammer detection in microblogging. In *IJCAI*, 2013.

C.L. Lawson and R.J. Hanson. *Solving least squares problems*, volume 15. SIAM, 1995.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Kyumin Lee, James Caverlee, and Steve Webb. Uncovering social spammers: social honeypots + machine learning. In *Proceedings of SIGIR*, 2010.

Kyumin Lee, Brian David Eoff, and James Caverlee. Seven months with the devils: A long-term study of content polluters on twitter. In *Proceedings of ICWSM*, 2011.

Yu-Ru Lin, Hari Sundaram, Yun Chi, Junichi Tatemura, and Belle L Tseng. Splog detection using self-similarity analysis on blog temporal dynamics. In *AirWeb*, 2007.

Omid Madani, Hung Hai Bui, and Eric Yeh. Efficient online learning and prediction of users' desktop actions. In *IJCAI*, 2009.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of ICML*, 2009.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online learning for matrix factorization and sparse coding. *The JMLR*, 2010.

D Seung and L Lee. Algorithms for non-negative matrix factorization. *NIPS*, 2001.

Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. User-level sentiment analysis incorporating social networks. In *Proceedings of KDD*, 2011.

Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu. Exploiting homophily effect for trust prediction. In *Proceedings of WSDM*, 2013.

K. Thomas, C. Grier, D. Song, and V. Paxson. Suspended accounts in retrospect: An analysis of twitter spam. In *Proceedings of ACM SIGCOMM conference on Internet measurement conference*, 2011.

Dong Wang, Ravichander Vipperla, Nicholas Evans, and Thomas Fang Zheng. Online non-negative convolutive pattern learning for speech signals. 2013.

S. Webb, J. Caverlee, and C. Pu. Social honeypots: Making friends with a spammer near you. In *Proceedings of CEAS*, 2008.

J. Weng, E.P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of WSDM*, 2010.

Haiqin Yang, Zenglin Xu, Irwin King, and Michael R Lyu. Online learning for group lasso. In *Proceedings of ICML*, 2010.

Z. Yang, C. Wilson, X. Wang, T. Gao, B.Y. Zhao, and Y. Dai. Uncovering social network sybils in the wild. In *Proceedings of IMC*, 2011.

K. Yu, S. Yu, and V. Tresp. Multi-label informed latent semantic indexing. In *Proceedings of SIGIR*, 2005.

D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of ICML*, 2005.

Y. Zhu, X. Wang, E. Zhong, N.N. Liu, H. Li, and Q. Yang. Discovering spammers in social networks. In *AAAI*, 2012.