Who will be Participating Next? Predicting the Participation of Dark Web Community

Xuning Tang
Drexel University
3141 Chestnut Street
Philadelphia, PA, USA
xt24@drexel.edu

Christopher C. Yang
Drexel University
3141 Chestnut Street
Philadelphia, PA, USA
chris.yang@drexel.edu

Mi Zhang
Drexel University
3141 Chestnut Street
Philadelphia, PA, USA
mz349@drexel.edu

ABSTRACT
Predicting whether a user will be participating in a thread has broad applications, such as thread recommendation and ranking. In an extremist forum, knowing which user will be interested to join a particular thread with sensitive or threatening information is also important for security agent to prevent or prepare for any potential outbreak of crisis. Traditional methods employed a bipartite graph to represent user-thread relationships and predict potential users for a new coming thread based on user similarities. In this paper, we propose a User Interest and Topic Detection model to extract topics and trends from a document corpus and also discover users' interests toward these trends. Information of user interest is then used to predict potential information consumers for a given thread. Experiments conducted in the Dark Web dataset showed the effectiveness of our approach; especially when we have limited information about who have already participated in an existing new thread.

Categories and Subject Descriptors
H.2.8 [**Database Management**]: Database Applications – Data Mining

General Terms
Algorithms, Experimentation

Keywords
Ranking, Topic and Trend Modeling, User Interest, Prediction

1. INTRODUCTION
Extremist forum is an ideal place to track and understand ongoing hot topics discussed by potential terrorists or extremists. In an online forum, every single word is recorded and the timestamp of interaction is captured, which is unimaginable in real world surveillance. Understanding the major concerns, arguments or conversations in extremist forums will offer great support to counterterrorism tasks and homeland security missions. In particular, whenever a thread of threatening topic, such as suicide bomber, is being discussed, it's important to understand which forum user might be interested to this topic. In that way, security agent can make better response to prevent or prepare for any potential outbreak of crisis.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ISI-KDD'12, August 12–16, 2012, Beijing, China.
Copyright 2010 ACM ISBN 978-1-4503-1550-0…$10.00.

An online forum consists of hundreds and thousands of threads and users. Each thread can be considered as a collaborative contribution of several users. An initiator posts an original message and the other commenters reply either to the original message or to each other's replies. Depending on its content, each thread might involve one or more different ***topics***, such as military topic, economy topic or counter-terrorism topic. Threads about common stories or events form a ***trend***, such as the Killing of Osama Bin Laden by U.S. Force, or the Withdrawal of U.S. Force from Iraq and Afghanistan. Users may follow different trends in an online forum and post their messages. On the other hand, online forum has its social function which offers a platform for its users to chat or socialize with each other. Some factors such as creator reputation and interestingness of participants may influence users' decisions on whether to participate in the thread discussions [3; 4]. Therefore, users participate in a thread not only because of the trend/topic/content of this thread, but also due to the other participants of this thread. In this work, we narrow down our research focus to predict potential users given a new coming thread in an online forum. Generally speaking, we employ a bipartite graph to represent the user-thread relationships and quantify user similarities based on the number of common threads that they have. In addition, we propose a User Interest and Topic Detection model to capture major topics and trends of a corpus of documents, and also model users' interests toward those detected trends. When a new thread is initiated, our User Interest and Topic Detection model will first classify it to one of existing trends and then return a list of users who showed their interests in this trend before. We then leverage the power of both user similarity and user interest to predict who might be interested to this new thread assuming that we already know a small percentage of participants. Experiments conducted in the dark web dataset showed the effectiveness of our approach.

In the following sections, we first discuss some major related works, define the problem formally and then propose our methods to address the research problem. Experiment results are reported and discussed in section 5.

2. RELATED WORKS
Potential user prediction for thread discussions resembles the problem of recommender system in e-commerce, which recommends products to potential users [13]. One of the most important approaches for recommendation is collaborative filtering, which is built on the assumption that a good way to find people's interests is to find other people who have similar interests [2]. In recommender systems, bipartite graph is widely used to represent the relations of users and products [6; 11]. Bipartite graph is two-mode network, with one type of nodes

representing users and another type of nodes representing products. A link only exists between different types of nodes, and it is created if the corresponding user selects the corresponding product.

With user-product interacting data in bipartite graph, collaborative filtering for recommendation can be viewed as link prediction problem [9]. Huang et al. summarized and compared six different linkage measures for collaborative filtering, including Common Neighbors, Jaccard's Coefficient, Adamic/Adar, Preferential Attachment, Graph Distance and Katz. There are two types of algorithms for collaborative filtering, which are user-based algorithm and item-based algorithm [8; 9]. The user-based algorithm constructs a matrix to represent user similarities based on their overlapping interactions. Products are recommended to a user according to other users' actions and similarities to that user. The item-based algorithm constructs a matrix to represent similarities of products. Products are recommended to a user if she has chosen other similar products. User-based algorithm is more suitable for user prediction of thread discussions, given some other users participated in a same thread. A lot of research has been carried out to improve collaborative filtering in recommender systems. Personalization and content analysis are taken into account in different improving algorithms [6; 8; 11; 12].

Fung et al. [5] applied collaborative filtering in prediction for online discussion participation. They built user-thread matrix based on the bipartite graph, and proposed Weighted Non-negative Matrix Factorization (WNMF) to find latent factors in the matrix. In their approach, the ideas of Zipf's laws and tf-idf were applied in WNMF to calculate pfidf scores for every pair of user and thread in the matrix. However, collaborative filtering has the problem that it ignores the content of posts and comments for user prediction. To solve this problem, Yano el al. [14] developed an approach with topic models to predict response to political blog posts. They combined topic models of LinkLDA and CommentLDA. To generate a thread in their approach, the topics, words of the post, commenters and words of the comments are drawn based on their multinomial distributions respectively.

Collaborative filtering and topic model can predict users from different perspectives. However, there is seldom research combining these two different approaches to predict users. In this work, we first use collaborative filtering as the baseline approach to predict thread users, and then develop topic models to boost the baseline approach.

3. PROBLEM DEFINITION

As motivated in the introduction, we formally define the research problem in this paper as follows: given a collection of threads, D, from an online forum, assuming each thread d ($d \in D$) consists of a bag of words $W_d = \{w_{d_1}, w_{d_2}, \dots, w_{d_M}\}$ and a group of users $P_d = \{p_{d_1}, p_{d_2}, \dots, p_{d_N}\}$, let $\overline{P_d}$ represents one subset of P_d and $\overline{\overline{P_d}}$ represents another subset of P_d, such that $\overline{P_d} \cap \overline{\overline{P_d}} = \emptyset \; and \; \overline{P_d} \cup \overline{\overline{P_d}} = P_d$, the research problem is defined as predicting $\overline{\overline{P_d}}$ based on the observed set of $\overline{P_d}$.

4. METHODOLOGY

In this section, we first propose a research framework under which we addressed the problem defined in section 3. We then introduce the baseline approach and discuss how to boost the baseline approach by incorporating user interest. Finally, we propose a User Interest and Topic Detection Model (UTD) to capture user interest which serves as the input for the User-Interest-Boosted Rank method.

4.1 Research Framework

In this work, we use a bipartite graph to represent the relationships between users and threads, as shown in figure 1. Nodes on the left in Fig. 1 represent forum users and nodes on the right represent forum threads. A user is connected to a thread by an edge if the user posted any message within this thread. To predict potential users given a new coming thread, we first calculate the similarities between all pairs of users according to the user-thread bipartite graph. Secondly, based on a small portion of users whom has already been known that they are participating in a new thread, we adopt a collaborative algorithm to rank all other users based on their similarities to this group of existing users. Finally, we train a UTD model which can assign a new thread to one of the existing trends and return a list of users with strong interest to this trend. This resulted list will then be used to boost the ranking generated based on user similarity. The whole procedure is depicted by Figure 2 below.

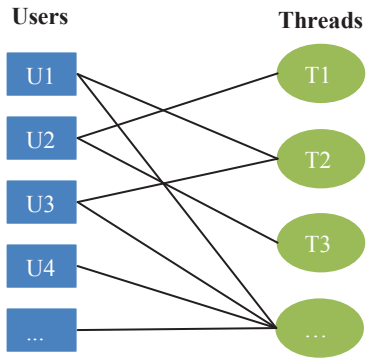


Figure 1. Bipartite Graph of Users and Threads

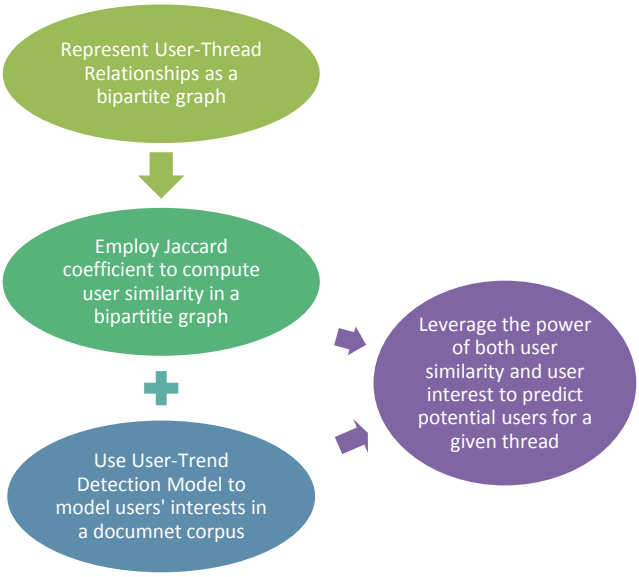


Figure 2. Research Framework

4.2 Baseline

As mentioned before, user-thread relationship is captured by a bipartite graph in this work. Therefore, the similarity between two users i and j is calculated based on the number of threads in which i and j coexisted. Define formally, let C_i denotes the set of threads

that user i participated in. The similarity between user i and j is defined by Jaccard index [7] as:

$$Sim(i,j) = \frac{|C_i \cap C_j|}{|C_i \cup C_j|} \quad (1)$$

Given a new thread d and a small set of users, $\overline{P_d}$, who are already in thread d, we rank users, excluding those within $\overline{P_d}$, based on $\Pr(\overline{\overline{P_{d_i}}}|\overline{P_d}, \mathcal{G})$ defined as:

$$\Pr(\overline{\overline{P_{d_i}}}|\overline{P_d}, \mathcal{G}) = \frac{\sum_{j \in \overline{P_d}} |\mathrm{Sim(i,j)}|}{|\overline{P_d}|} \quad (2)$$

where $\overline{\overline{P_{d_i}}}$ represents a user i who has not participated in thread d yet, and $\mathcal{G}$ represents a bipartite graph. This ranking method relying on user similarity will serve as the baseline approach in this work.

4.3 User-Interest-Boosted Rank (UB-Rank)

In this section, we introduce a simple but effective way to boost the baseline ranking method. The rationale behind this user-interest-boosted rank is straightforward. In the baseline ranking method, we neglect the content information of a new thread, e.g. the topics of this new thread and the trend this new thread may belong to, but only consider the user similarity based on their previous participations. In the user-interest-boosted rank, we take the users' interests toward trends and topics into consideration in addition to the user similarity. In other words, it may help us to better predict potential thread participants if we can acquire topic and trend information of a new thread as well as users' interests to these topics and trends.

Assuming we already know that a new thread d belongs to a trend c_d, and also assuming that we use a vector $\Omega_{c_d} = \{\Omega_{c_d,1}, \Omega_{c_d,2}, \dots, \Omega_{c_d,u}\}$ to represent users' interests to trend c_d, where $\Omega_{c_d,i}$ denotes user $i's$ interest to trend c_d, the boosted ranking score for user $\overline{\overline{P_{d_i}}}$ to document d, $\Pr(\overline{\overline{P_{d_i}}}|\overline{P_d}, \mathcal{G}, c_d, \Omega_{c_d})$, is then redefined as:

$$\Pr(\overline{\overline{P_{d_i}}}|\overline{P_d}, \mathcal{G}, c_d, \Omega_{c_d}), = w \times \Pr(\overline{\overline{P_{d_i}}}|c_d, \Omega_{c_d}) + (1-w) \times \Pr(\overline{\overline{P_{d_i}}}|\overline{P_d}, \mathcal{G}) \quad (3)$$

where w is a weight, $\Pr(\overline{\overline{P_{di}}}|c_d, \Omega_{c_d})$ corresponds to user $i's$ interest to trend c_d which can be captured by our UTD Model discussed below.

4.4 User Interest Modeling

There are several different ways to capture users' interests. A naïve approach is to collect all texts (both posts and comments) posted by a user and then aggregate them into a fix-length TF-IDF vector to be a "word profile" of a user. The interest of a user to a document is then measured by the similarity between the given document and this user's word profile. However, this simple method only considers word similarity which is merely a single dimension. In this paper, we propose to extract trends and topics from a document corpus, which have much richer semantic meaning than plain words, and then capture users' interests toward trends.

4.4.1 User Interest and Topic Detection Model

Our User Interest and Topic Detection Model (UTD) is inspired by both Blei's LDA model [1] and Kawamae's Trend Analysis Model [10]. General speaking, Blei et al. proposed a LDA model which can extract topics from a document corpus given an assumption that each document contains different topics and each word is generated by one of these topics. A topic is therefore considered as a cluster of words that frequently co-occur. Kawamae extended the traditional LDA model by introducing a latent trend class variable into each document so that each document belongs to one trend and each trend consists of topics. Our UTD model further extends the Trend Analysis Model by taking into account users' interest in the generation process of a document corpus. In this work, we do not consider the timestamp of a document but can be easily extended in the future work.

Our UTD model is designed based on a real generation process of terms and users. In an online forum, we consider each thread as a document. Content terms of a document are composed of the terms from both the initial post of a thread and all of its comments. Similarly, the initiator of a post and all commenters of a post are considered as the users of this document. The generation process of each document d in our UTD model is as follows: first of all, UTD determines a trend class label c_d for document d. Once c_d is determined, users' interests play an important role to decide whom this new document will attract. UTD model will then sample $|U_d|$ users, each from a multinomial distribution with a Dirichlet prior depending on c_d. At third, UTD generates content terms for a document following a LDA-like process. Concretely, we assume that the generation of each term of d is influenced by one of three factors: 1) a general background topic; 2) a trend-background topic (e.g. Killing of Osama Bin Laden); 3) topics belonging to the whole corpus (e.g. counter-terrorism, military operation and etc.). Terms that are topic-unrelated but widely exist in the corpus have higher chance coming from the general background topic. In addition, each trend may contain some signature terms which serve as the trend-background terms. At last, similar to LDA model, the whole corpus shares a mixture of |Z| different topics. We adopt a switch variable to control the influence of these three factors on content term generation. The proposed model is shown in Figure 3. The meanings of notations used in Fig. 3 are listed in table 1.

As shown in figure 3, the trend class label c_d of document d is sampled by a multinomial distribution φ with a Dirichlet prior α. Once c_d is determined, let $|U_d|$ equal to the number of users participated in document d, the user list of d is generated by repeating the sampling process $|U_d|$ times based on the trend-user distribution Ω_{c_d}. Each word w_{d_i} of d is drawn from either the general background topic or the trend-background topic or one of |Z| topics shared by the whole corpus. As mentioned, a switch variable x_{d_i} is first drawn from a multinomial distribution for word $w_{d_i}$to control its generation process. If $x_{d_i} = 0$, w_{d_i} is drawn from the general background topic, ϕ_{bg}. If $x_{d_i} = 1$, w_{d_i} is drawn from the trend-background topic, ϕ_{c_d}, which consists of the signature terms of a trend. If $x_{d_i} = 2$, a topic z_{d_i} is first sampled from the trend-topic distribution θ_{c_d}, and then the word w_{d_i} is drawn conditionally on the sampled topic. Overall, the generation process of users and words in the UTD model can be described as follows:

1. Draw 1+C+Z multinomial distributions ϕ_z from prior γ, one for each topic (1 general background topic, C trend-specific background topics and Z other topics)
2. Draw D multinomial distribution μ_d from prior ϵ, one for each document d;
3. Draw C multinomial distributions Ω_c and θ_c from prior λ and β respectively, one for each trend class c;

For each document d:

A. Draw an trend class label c_d from φ
B. Draw $|U_d|$ users from multinomial distribution Ω_{c_d}

For each content term w_{d_i} in document d:

Draw a switch variable x_{d_i} from multinomial μ_d

if $x_{d_i} = 0$

i. Draw word w_{d_i} from multinomial ϕ_{bg}

else if $x_{d_i} = 1$

i. Draw word w_{d_i} from multinomial ϕ_{c_d}

else if $x_{d_i} = 2$

i. Draw topic z_{d_i} from multinomial θ_{c_d}

ii. Draw word w_{d_i} from multinomial $\phi_{z_{d_i}}$

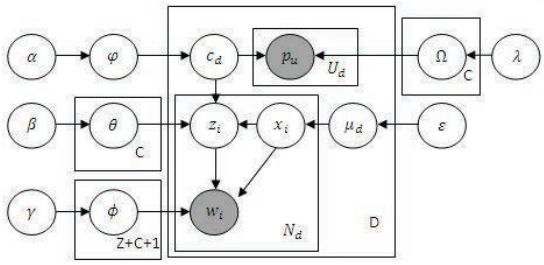


Figure 3. Graphical Model of User Interest and Topic Detection Model

Table 1. Notations

C, c_d	Trend labels of all documents, the trend label of document d
P, P_u	All users , user u
Z, z_{d_i}	All topics , the topic of the i^{th} word in document d
W, w_{d_i}	All words , the i^{th} word of document d
X, x_{d_i}	All switch variables , the switch variable of the i^{th} word in document d
$\alpha, \gamma, \beta, \varepsilon, \lambda$	parameters of symmetric Dirichlet priors
φ	the multinomial distribution of trend class
ϕ_z	the multinomial distribution of words specific to topic z
Ω_c	the multinomial distribution of users specific to trend c
θ_c	the multinomial distribution of topics specific to trend c
μ_d	the multinomial distribution of switch variables specific to document d
$X_{d,x=0,1,2}$	number of word tokens assigned to switch x = 0/1/2
$\mathrm{M_{z,w}}$	number of times that word w is assigned to topic z
$\mathrm{K_c}$	number of times that a document is assigned to trend c
$\mathrm{N}_{c,z}$	number of times that topic z is assigned to trend c
$\mathrm{L_{c,u}}$	number of times that user u is assigned to trend c

4.4.2 Inference and Learning

As shown in Figure 3, the shaded nodes P_u and w_{d_i} are observed variables; $\alpha, \gamma, \beta, \varepsilon$ *and* λ are Dirichlet priors which can be predefined; $\varphi, \phi, \Omega, \theta$ *and* μ can be eventually integrated out, so that c_d, z_{di} *and* x_{d_i} are the only hidden variables need to estimate based on the observations. Gibbs sampling is an effective way to estimate these hidden variables. In Gibbs sampling, Markov chains are constructed to approximate the conditional distribution of those hidden variables. The key of the Gibbs sampling algorithm for our UTD model is to approximate the conditional distribution of trend class c_d, topic z_{di} and switch variable x_{d_i}. Therefore, the sampling scheme consists of two steps. The first step is to sample the trend label c_d for document d, $Pr(c_d = c| \dots)$. The second step is to sample the switch variables and the topics for each individual word of this document: $\Pr(x_{di} = 0| \dots)$, $\Pr(x_{di} = 1| \dots)$ and $\Pr(z_{di} = k, x_{di} = 2| \dots)$. We first derive the joint distribution of the entire corpus and then introduce the conditional probability of the hidden variables for Gibbs sampling.

4.4.2.1 Joint Distribution

As shown in the previous subsection, the joint distribution of the entire corpus is computed as:

$$\Pr(C, P, Z, W, X|\alpha, \gamma, \lambda, \beta, \varepsilon)$$
$$= \int\int\int\int\int \Pr(C, P, Z, W, X, \phi, \Omega, \theta, \varphi, \mu|\alpha, \gamma, \lambda, \beta, \varepsilon) d\mathcal{H}$$
$$= \int\int\int\int\int \prod_d^D [\Pr(c_d|\varphi) \prod_u^{U_d} \Pr(P_u|\Omega_{c_d})$$
$$\times \prod_i^{N_d} (\Pr(x_{d_i}|\mu_d) \Pr(z_{d_i}|x_{d_i}, \theta_{c_d}) \Pr(w_{d_i}|\phi_{z_{d_i}}))] \cdot \Pr(\varphi|\alpha)$$
$$\times \prod_d^D \Pr(\mu_d|\varepsilon) \prod_z^{Z+C+1} \Pr(\phi_z|\gamma) \prod_c^C [\Pr(\Omega_c|\lambda) \Pr(\theta_c|\beta)] \, d\mathcal{H} \quad (4)$$

where $d\mathcal{H} = \mathrm{d}\phi d\Omega d\theta d\varphi d\mu$. In the joint distribution above, multinomial distributions $(\varphi, \phi, \Omega, \theta, \mu)$ can be adapted by the conjugate priors $(\alpha, \gamma, \lambda, \beta, \varepsilon)$ and then integrated out eventually.

4.4.2.2 Trend Class Label Sampling

In the sampling schema, for each document, we use the chain rule to obtain the conditional distribution $\Pr(c_d = c| \dots)$ as:

$$\Pr(c_d = c| \dots) = \frac{\mathrm{K_c} + \alpha_c - 1}{\sum_c^C (\mathrm{K_c} + \alpha_c) - 1} \cdot \left[\frac{\prod_u^U \Gamma(\mathrm{L_{c,u}} + \lambda_u)}{\prod_u^U \Gamma(\mathrm{L_{c,u\backslash d}} + \lambda_u)} \cdot \frac{\Gamma(\sum_u^U (\mathrm{L_{c,u\backslash d}} + \lambda_u))}{\Gamma(\sum_u^U (\mathrm{L_{c,u}} + \lambda_u))} \right]$$
$$\times \left[\frac{\prod_z^Z \Gamma(\mathrm{N_{c,z}} + \beta_z)}{\prod_z^Z \Gamma(\mathrm{N_{c,z\backslash d}} + \beta_z)} \cdot \frac{\Gamma(\sum_z^Z (\mathrm{N_{c,z\backslash d}} + \beta_z))}{\Gamma(\sum_z^Z (\mathrm{N_{c,z}} + \beta_z))} \right] \quad (5)$$

where $L_{c,u\backslash d}$ represents the number of times that user u has been assigned to trend c, except for document d. $N_{c,z\backslash d}$ represents the number of times that topic z is assigned to trend c, except for document d. The first term on the right handle of the above formula measures the probability of assigning this document to trend c; the second term (within the 1st square bracket) measures the probability of observing the users in document d given d belongs to trend c; and the third term (within the 2nd square bracket) measures the probability of observing the topics given d belongs to trend c.

4.4.2.3 Switch Variable and Topic Sampling

For each word token, the posterior probability of adding word w_{d_i} in document d to background topic is defined as:

$$\Pr(x_{-di} = 0| \dots) \propto \frac{X_{d,0} + \varepsilon_0 - 1}{\sum_x^X (\mathrm{X_{d,x}} + \varepsilon_x) - 1} \cdot \frac{\mathrm{M_{z=bg,w}} + \gamma_w - 1}{\sum_w^W (\mathrm{M_{z,w}} + \gamma_w) - 1} \quad (6)$$

where the first term, $\frac{X_{d,0} + \varepsilon_0 - 1}{\sum_x^X (\mathrm{X_{d,x}} + \varepsilon_x) - 1}$, measures the probability of having switch variables equals to 0, and the second term, $\frac{\mathrm{M_{z=bg,w}} + \gamma_w - 1}{\sum_w^W (\mathrm{M_{z,w}} + \gamma_w) - 1}$, measures the probability of generating w_{d_i} from a background topic.

Similarly, the posterior probability of adding word w_{d_i} in document d to trend-specific background topic c is defined as:

$$\Pr(x_{-di} = 1| \dots) \propto \frac{X_{d,1} + \varepsilon_1 - 1}{\sum_x^X (\mathrm{X_{d,x}} + \varepsilon_x) - 1} \cdot \frac{\mathrm{M_{z=c,w}} + \gamma_w - 1}{\sum_w^W (\mathrm{M_{z,w}} + \gamma_w) - 1} \quad (7)$$

where the first term measures the probability of having switch variables equals to 1, and the second term measures the probability of generating w_{d_i} from the trend-background topic.

Similarly, the posterior probability of adding word w_{d_i} in document d to topic k is defined as:

$$\Pr(z_{-di} = k, x_{-di} = 2| \ldots) \propto \frac{X_{d,2}+\varepsilon_2-1}{\sum_x^X(X_{d,x}+\varepsilon_x)-1} \cdot \frac{N_{c,k}+\beta_k-1}{\sum_z^Z(N_{c,z}+\beta_z)-1} \cdot \frac{M_{z=k,w}+\gamma_w-1}{\sum_w^W(M_{z,w}+\gamma_w)-1} \quad (8)$$

where the first term measures the probability of having switch variables equals to 2, the second term measures the probability of selecting topic k in trend c, and the third term measures the probability of generating w_{d_i} from topic k.

4.4.2.4 Parameter Estimation

Once the sampling process based on the posterior distributions calculated in sections 4.4.2.2 and 4.4.2.3 converged, we can estimate the five parameters using the following equations:

$$\theta_{c,k} = \frac{N_{c,k}+\beta_k}{\sum_z^Z(N_{c,z}+\beta_z)}, \phi_{z,w} = \frac{M_{z,w}+\gamma_w}{\sum_w^W(M_{z,w}+\gamma_w)}, \varphi_c = \frac{K_c+\alpha_c}{\sum_c^C(K_c+\alpha_c)}$$

$$\mu_{d,x} = \frac{X_{d,x}+\varepsilon_x}{\sum_x^X(X_{d,x}+\varepsilon_x)}, \Omega_{c,u} = \frac{L_{c,u}+\lambda_u}{\sum_u^U(L_{c,u}+\lambda_u)}$$

Among these variables, we have particular interest in $\Omega_{c,u}$ which denotes a trend-user distribution. In other words, $\Omega_{c,u}$ can be used to represent user u's interest toward trend c. For each trend c, we can rank all users based on $\Omega_{c,u}$, the higher the rank is, the larger the chance that a user will be interested to documents of trend c.

5. EXPERIMENT

5.1 Dataset

The dataset provided by ISI-KDD 2012 Challenge consists of messages from a Dark Web Portal which offers access to several multi-year extremist forums. In total, there are 27,968 threads, 129,425 comments and 2,803 unique users. We remove users with very few contributed messages as well as their messages since these users are not core users and predicting their participations is unimportant. After filtering, there are 1,360 users, 27,063 threads and 123,754 comments in the dataset.

Our proposed UTD model treats each thread as a document. For each thread, we extracted all terms in its initial post and following comments. The data is preprocessed by discarding non-alphabetic content, removing general stop words, stemming and removing low-frequency words.

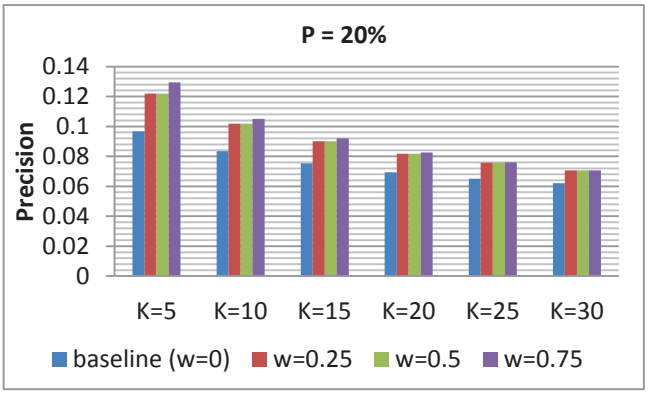
We randomly split the dataset into training set and test set, which contains 24,416 and 2,550 threads respectively. To guarantee that threads in the test set have enough users to evaluate the prediction method, all threads in the test set have at least five users.

The training set contains the full information for each single thread, including all users and content terms, which we used to train our proposed model. The test set was used to evaluate the prediction algorithms. For each thread d in the test set, we randomly removed some users. Let $\overline{P_d}$ be the set of users that are retained, which means we know their participation in d. Let $\overline{\overline{P_d}}$ be the set of users that are removed, which the prediction algorithms try to predict. Let $P = |\overline{P_d}|/(|\overline{P_d}| + |\overline{\overline{P_d}}|)$ and we generated three test sets with P=20%, P=40% and P=60%, respectively.

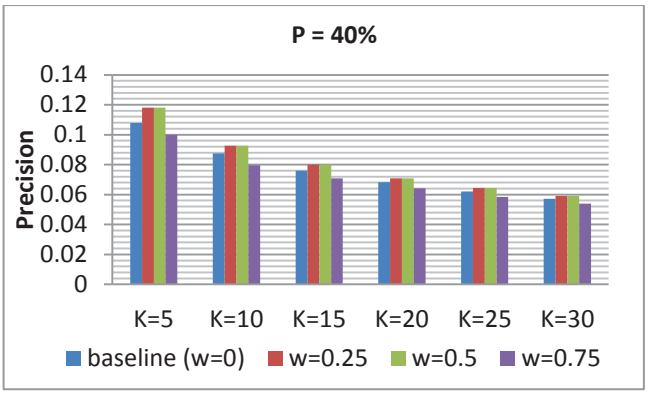
5.2 Evaluation Criteria

Precision@K is used to evaluate the results of our experiment. Specifically, by using Precision@K, we consider the top-K list returned by a prediction method. Let m denotes the number of users that co-exist in the top-K list and the ground truth participant list of a thread, Precision@k equals to m divided by K. For each test sets with different P, the experiment was repeated five times and only the average Precision@K were reported.

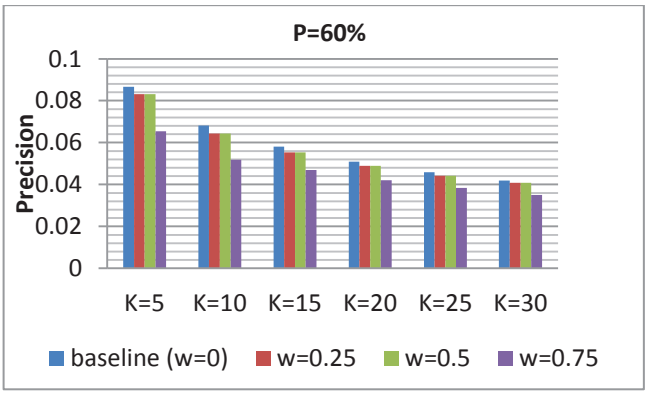
5.3 Experiment Results and Discussions



(a)



(b)



(c)

Figure 4. Precision @ K of User Prediction in Dark Web Dataset

The experiment setup is as follows: 1) Using a training set, we constructed a bipartite graph and calculated user similarities; 2) Based on the user similarities computed in step 1, we applied the baseline ranking method (in Eq. 2) to predict users for each thread

in the test set; 3) Using the training set again, we applied our proposed UTD model to discover users' interests to trends; 4) We applied the trained UTD model to the test set to assign a trend label to each thread in the test set; 5) For each thread in the test set, since we knew its trend label from step 4 and also users' interests to each trend from step 3, we applied our boosted method (in Eq. 3) to predict its potential users.

Using the method introduced in section 4.2 as baseline, we tested the performance of our proposed UB-Rank with different settings of parameters. It's important to note that, when w=0, it becomes the baseline approach which relies only on bipartite graph structure. Two major parameters that we tested in this experiment are P and w. P represents the percentage of users that we know for a thread in a test set. *w* is the weight in equation (3). Figure 4(a) demonstrated that when P = 20%, our proposed UB-Rank substantially outperformed the baseline approach especially when w = 0.75. Specifically, when P = 20% and w = 0.75, in terms of Precision@5, UB-Rank outperformed the baseline by 34% (0.129 vs. 0.096). The improvement decreases a bit when we consider a larger value of K. But UB-Rank with w = 0.75 still outperformed baseline by 14% in terms of Precision@30. When we decrease w from 0.75 to 0.25, which makes the UB-Rank replying more on user similarity rather than user interest, we observed a lower improvement. But UB-Rank with w = 0.25 still outperformed baseline by 26% in terms of Precision@5 and 14% in terms of Precision@30.

Figure 4(b) shows that when P = 40%, which means we know 40% of the users in a new thread, UB-Rank can still outperform baseline when w = 0.25 and w = 0.5. The improvement is larger when K is smaller. However, when P = 40%, UB-Rank with w = 0.75 was not comparable to the baseline. This means that when we know more users in a thread, the improvement by inducing user interest is less significant. This phenomenon becomes more obvious when P increase to 60% in Figure 4(c), where UB-Rank is less effective to predict potential users than purely using user similarity.

The experiment results above showed that our proposed method can effectively capture user interest. User interest is very useful information to predict potential users in a forum, especially when only a small percentage of users are observed for a thread. In this case, our model can find out which trend this new thread belongs to and take advantage of users' interest toward this trend to boost the baseline ranking. However, when more users are known in a thread, UB-Rank doesn't perform so well. A possible explanation is that: a thread attracts users because of two reasons, interestingness of content and interestingness of participants [4]. In an initial post which only has a few participants, the content plays a more important role to attract user. As a result, if we can detect which trend this new thread might belong to and who are the users feeling interested to this trend, it can help us to better predict potential users. On the other hand, in a post which already had a lot of participants, content interestingness may be less important. Users might join a thread because they see other interesting participants, such as people that they are familiar with or people with good reputation, without caring too much about the content. This is particularly important if the Web forum or social media site has the social function such as "follow" or "like" so that users may be alerted when the interesting participants they are following are replying to a thread. Although the performance of our proposed method declines when P is over 60%, it is more important to predict potential users when P is low which happens at the beginning stage of a thread. In other words, if we already know most of the users in a thread or the thread has been carried on for a period of time, the value of predicting the rest of the users is relatively low.

6. CONCLUSION

In this paper, we addressed the research problem of predicting potential information consumer for a given thread in an online forum. By knowing which users will be interested to join some threads with sensitive issues, security agents can prevent or prepare for potential outbreak crisis. Specifically, we quantified the similarity of pairs of users based on the number of common threads joined together by these users. Further, we proposed a User Interest and Topic Detection Model which can extract topics and trends from document corpus and can quantify users' interests toward each detected trend. This information of user interest is then used to boost the ranking generated only based on user similarity using collaborative filtering method. Experiment results demonstrated that when a small portion of users are observed in a new thread, our approach can substantially improve the precision of information consumer prediction.

7. REFERENCES

[1] BLEI, D.M., NG, A.Y., and JORDAN, M.I., 2003. Latent Dirichlet Allocation. *The Journal of Machine Learning Research 3*(Mar. 2003), 993-1022. DOI= http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993.

[2] BREESE, J.S., HECKERMAN, D., and KADIE, C., 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the the Fourteenth conference on Uncertainty in artificial intelligence* (Madison, WI, July 24-26 1998), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 43-52.

[3] CHOUDHURY, M.D. and SUNDARAM, H., 2011. Why do we converse on social media?: an analysis of intrinsic and extrinsic network factors. In *Proceedings of the the 3rd ACM SIGMM international workshop on Social media* (Scottsdale, AZ, USA, November 28 - December 01 2011), ACM, New York, NY, USA, 53-58. DOI= http://dx.doi.org/10.1145/2072609.2072625.

[4] CHOUDHURY, M.D., SUNDARAM, H., JOHN, A., and SELIGMANN, D.D., 2009. What makes conversations interesting?: themes, participants and consequences of conversations in online social media. In *Proceedings of the the 18th international conference on World wide web* (Madrid, Spain, April 20-24 2009), ACM, New York, NY, USA, 331-340. DOI= http://dx.doi.org/10.1145/1526709.1526754.

[5] FUNG, Y.-H., LI, C.-H., and CHEUNG, W.K., 2007 Online Discussion Participation Prediction Using Non-negative Matrix Factorization. In *Proceedings of the the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops* (Silicon Valley, California, USA, November 2-5 2007), IEEE Computer Society, Washington, DC, USA, 284-287. DOI= http://dx.doi.org/10.1109/WI-IATW.2007.128.

[6] GRUJIĆ, J., 2008. Movies Recommendation Networks as Bipartite Graphs. In *Proceedings of the 8th International Conference on Computer Science* (Kraków, Poland, June 23-25 2008), Springer, Verlag Berlin, Heidelberg, 576–583. DOI= http://dx.doi.org/10.1007/978-3-540-69389-5.

[7] HAMERS, L., HEMERYCK, Y., HERWEYERS, G., JANSSEN, M., KETERS, H., ROUSSEAU, R., and VANHOUTTE, A., 1989. Similarity measures in scientometric research: the Jaccard index versus Salton's cosine formula. *Information Processing and Management:*

an International Journal 25, 3 (May 1989), 315-318. DOI= http://dx.doi.org/10.1016/0306-4573(89)90048-4.

[8] HUANG, Z., 2007. Selectively acquiring ratings for product recommendation. In *Proceedings of the the ninth international conference on Electronic commerce* (Minneapolis, MN, USA, August 19-22 2007), ACM, New York, NY, USA, 379-388. DOI= http://dx.doi.org/10.1145/1282100.1282171.

[9] HUANG, Z., LI, X., and CHEN, H., 2005. Link prediction approach to collaborative filtering. In *Proceedings of the Joint Conference on Digital Libraries 2005* (Denver, CO, USA, June 07-11 2005), ACM, New York, NY, USA, 141-142. DOI= http://dx.doi.org/10.1145/1065385.1065415.

[10] KAWAMAE, N., 2011. Trend analysis model: trend consists of temporal words, topics, and timestamps. In *Proceedings of the the fourth ACM international conference on Web search and data mining* (Kowloon, Hong Kong, February 09-12 2011), ACM, New York, NY, USA, 317-326. DOI= http://dx.doi.org/10.1145/1935826.1935880.

[11] LIU, J.-G., ZHOU, T., WANG, B.-H., ZHANG, Y.-C., and GUO, Q., 2010. Degree correlation of bipartite network on personalized recommendation. *International Journal of Modern Physics C 21*, 1, 137-147. DOI= http://dx.doi.org/10.1142/S0129183110014999.

[12] LIU, R.-R., LIU, J.-G., JIA, C.-X., and WANG, B.-H., 2010. Personal recommendation via unequal resource allocation on bipartite networks *Physica A: Statistical Mechanics and its Applications 389*, 16, 3282-3289. DOI= http://dx.doi.org/http://dx.doi.org/10.1016/j.physa.2010.04.004.

[13] RESNICK, P. and VARIAN, H.R., 1997. recommender systems. *Communications of the ACM 40*, 3 (Mar. 1997), 56-58. DOI= http://dx.doi.org/10.1145/245108.245121.

[14] YANO, T., COHEN, W.W., and SMITH, N.A., 2009. Predicting response to political blog posts with topic models. In *Proceedings of the Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (Boulder, Colorado, May 31-June 5 2009), Association for Computational Linguistics, Stroudsburg, PA, USA, 477-485.