Incorporating Social Context and Domain Knowledge for Entity Recognition

Jie Tang[†♯], Zhanpeng Fang[†], and Jimeng Sun[‡]
[†]Department of Computer Science and Technology, Tsinghua University
[♯]Tsinghua National Laboratory for Information Science and Technology (TNList)
[‡]College of Computing, Georgia Institute of Technology, USA
jietang@tsinghua.edu.cn, fzp13@mails.tsinghua.edu.cn, jsun@cc.gatech.edu

ABSTRACT

Recognizing entity instances in documents according to a knowledge base is a fundamental problem in many data mining applications. The problem is extremely challenging for short documents in complex domains such as social media and biomedical domains. Large concept spaces and instance ambiguity are key issues that need to be addressed.

Most of the documents are created in a social context by common authors via social interactions, such as *reply* and *citations*. Such social contexts are largely ignored in the instance-recognition literature. How can users' interactions help entity instance recognition? How can the social context be modeled so as to resolve the ambiguity of different instances?

In this paper, we propose the SOCINST model to formalize the problem into a probabilistic model. Given a set of short documents (e.g., tweets or paper abstracts) posted by users who may connect with each other, SOCINST can automatically construct a context of subtopics for each instance, with each subtopic representing one possible meaning of the instance. The model is also able to incorporate social relationships between users to help build social context. We further incorporate domain knowledge into the model using a Dirichlet tree distribution.

We evaluate the proposed model on three different genres of datasets: ICDM'12 Contest, Weibo, and I2B2. In ICDM'12 Contest, the proposed model clearly outperforms (+21.4%; $p \ll 1e-5$ with t-test) all the top contestants. In Weibo and I2B2, our results also show that the recognition accuracy of SOCINST is up to 5.3-26.6% better than those of several alternative methods.

Categories and Subject Descriptors

J.4 [**Social Behavioral Sciences**]: Miscellaneous; H.3.3 [**Information Search and Retrieval**]: Text Mining

General Terms

Algorithms, Experimentation

Keywords

Instance recognition; Social network; Probabilistic model

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741135 .

1. INTRODUCTION

The rapid development of online social networks has significantly enriched our daily communications. People use blogs, forums, and product review sites to share opinions on topics such as politicians or experiences with products, and also use the online space as the main channel to acquire and share information. As a result, large volumes of unstructured short-text documents are created. One fundamental analytic operation is to recognize entity instances in the document that map to existing concepts in a knowledge base. In social media, when talking about a specific product, different users may use different words when they mention the product (e.g., S4 vs. Samsung Galaxy S4). In some cases, people may deliberately use some new words to name an event or a product, just for fun or for other purposes, such as using "Fruit company" to name Apple Inc. or "Peace West King" to refer to "Xilai Bo" (a sensitive Chinese politician). In the medical domain, we may want to identify diverse instances from a medical corpus such as PubMed abstracts that map to an existing medical knowledge base such as the Unified Medical Language System (UMLS)[1].

The problem of entity instance recognition and linking requires simultaneous entity recognition and entity matching, which is a much harder problem than either one by itself. Despite many studies on related topics including entity recognition [9, 25, 13, 23, 27], entity matching [2, 3, 21, 29], entity resolution [4, 18, 22, 32], and entity morph [17] such combination and leveraging of social context are largely absent in the literature. Specifically, most entity recognition methods treat messages independently, without considering shared social context. Entity matching and resolution aim at linking entities from different sources with the same meaning. Entity morph is a special case of entity resolution, with an emphasis on connecting those instances intentionally concealed by users, for example, for avoiding censorship. Unlike this paper, none of them explicitly consider the structure of the social network and external knowledge base.

Figure 1 illustrates the problem addressed in this paper. The left figure shows three users with friend relationships, and the middle shows one original microblog posted by user A, and one reply to the original microblog by B, and one retweeted by C. The underlined text indicates three instances recognized by the proposed model and linked to the corresponding concepts in the knowledge base. The problem is non-trivial. First, as each message is very short, traditional methods for entity recognition that deal with messages independently would lead to unsatisfactory results. Second, the social network provides rich structure information, which should be leveraged. Third, the knowledge base also encodes useful domain knowledge (e.g., hypernyms and hyponyms), which should be utilized in the instance recognition process.

[1]http://www.nlm.nih.gov/research/umls

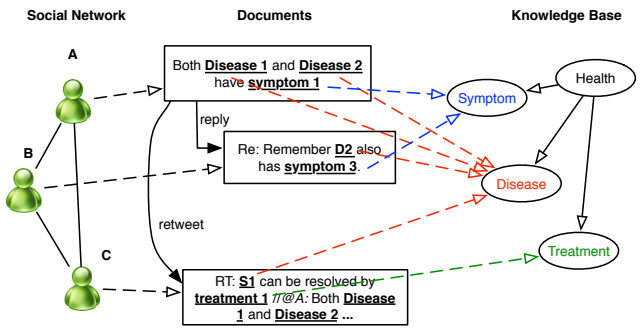


Figure 1: Example of entity instance recognition and linking in a microblogging network. The left figure shows three users with friend relationships, and the middle shows one original microblog from user A, and one reply from B, and a retweet from C. The underlined text indicates three instances recognized by the proposed model and linked to the corresponding concepts in the knowledge base

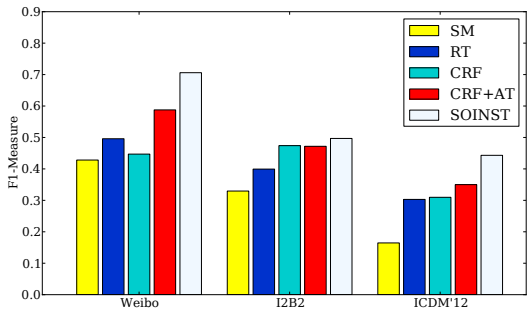


Figure 2: Performance comparison of SOCINST with comparative algorithms (in terms of F1-score). Please refer to § 5.1 for definitions of the comparative algorithms

In this paper, we study the problem of instance recognition and linking in a social context. We propose an SOCINST model to formalize the problem as a probabilistic topic model. Given a training set, the SOCINST method can automatically construct a context of subtopics for each instance by leveraging social relationships between users, with each subtopic representing one potential meaning of the instance. We further use a Dirichlet tree distribution to incorporate domain knowledge from the knowledge base into the topic model. We test the proposed model on three different genres of datasets: ICDM'12 Contest, Weibo, and I2B2. Our experimental results validate the effectiveness of the proposed SOCINST model in all three datasets. Our method improves the recognition accuracy by up to 5.3-26.6% compared with several alternative methods. Figure 2 shows the performance comparison of different methods on the three different datasets. Clearly, SOCINST performs much better than the other comparative algorithms. As an example, on the ICDM'12 Contest dataset, our method significantly outperforms (+21.4%; $p \ll 1e-5$ with t-test) the performance of the first place winner in the contest.

Organization. Section 2 formulates the problem; Section 3 enumerates preliminary considerations; Section 4 explains the proposed model and describes the algorithm for learning it; Section 5 presents the experimental results; Section 6 discusses related work and Section 7 concludes the work.

2. PROBLEM DEFINITION

We first provide necessary definitions and then formally formulate the problem.

Definition 1. **Social Network.** Let $G = (V, E)$ denote the social network, where V is a set of users and $E \subset V \times V$ is a set of relationships between users. We use $v \in V$ to represent a user and $e_{ij} \in E$ to represent a social relationship between users v_i and v_j.

Let D denote a set of M documents authored by users from V, each document $d \in D$ containing a vector $\mathbf{w}_d$ of N_d words, in which each $w_{di} \in \mathbf{w}_d$ is chosen from a vocabulary of size W. The authors of each document is denoted as $V_d \subset V$. The set of documents authored by v is denoted as D_v. Documents can also have links with each other. For example, in PubMed, the link between documents could be citation; in Weibo, the link between documents (tweets) can be retweet (or reply). For easy explanation of the algorithm, we use $IN(d)$ to indicate the subset of documents that have links to d and $OUT(d)$ to indicate the subset of documents to which document d has links. Moreover, we give a simple definition of the knowledge base.

Definition 2. **Knowledge Base.** A knowledge base is represented as a triple $KB = (C, R, X)$, where C represents a set of concepts; R represents a set of relations between concepts; and X represents a set of instances of those concepts.

In our problem, the instances X are to be recognized from the free text. The definition is a brief version of the definition of a knowledge base [10, 36]. In a general setting, relations fall into two broad types: *Taxonomies*, that organize concepts into sub- or super-concept hierarchy; and *Associative relations*, that relate concepts rather than the sub- or super-concepts. In this work, we mainly consider the taxonomy relation — i.e., sub-concept and super-concept relations — but the proposed model in the following section can be essentially extended to model the associative relations.

Instance recognition involves identifying instances X from free text, according to concepts C defined in the knowledge base, and then populating the knowledge base KB with the recognized instances X. Here, each instance $x_k \in X$ can be either a word or a phrase (multiple consecutive words). However, as the instances might be represented in different forms with different keywords, it is important to leverage the context information to aid in the recognition process. In this paper, we consider two types of context information — *social context* and *domain knowledge*. For a user, social context denotes information represented by friends or communities with which the user has been involved. Domain knowledge indicates information encoded in the knowledge base or information encoded in related documents. More specifically, we use topic models to model the context information.

Definition 3. **Topic models.** A topic model θ_i of a user v_i is a multinomial distribution of words $P(w|\theta_i)$. The assumption behind it is that words are sampled following a distribution corresponding to the user.

Thus the social context of user v_i can be represented as a mixture of user-specific topic models $\sum_{j \in NB(v_i)} \gamma\theta_j$, where $NB(v_i)$ is the set of neighbors of v_i and γ is the weight of the corresponding distribution. In practice, γ can be defined in different ways, for example, as the strength of the relationship between users v_i and v_j. Analogously, we can define the context for a concept as a mixture of instance-specific distributions. Given these definitions, we can formally formulate the problem studied in this work.

Table 1: Notations

SYMBOL	DESCRIPTION
K	Number of topics
W	Number of unique words
N_d	Number of words in document d
v_d	The author of document d
$\mathbf{w}_d$	Vector form of words in document d
w_{di}	The i-th word in document d
c^j_{di}	The corresponding concept at the j-th level in KB for word w_{di}
z_{di}	Topic sampled for the i-th word in document d
θ_v	Multinomial distribution over topics specific to author v
ϕ_z	Multinomial distribution of words specific to topic z
α	Dirichlet prior to multinomial distributions θ
β, η	Dirichlet (tree) prior to multinomial ϕ

Problem: The input of our problem consists of a labeled training set $\{(\mathbf{w}_d, \mathbf{y}_d)\}$, a social network G, and a knowledge base KB. The training set corresponds to a set of document $D = \mathbf{w}_d$ and the instance labeling results $\mathbf{y}_d$ for each document d (represented as $\mathbf{w}_d$), where $\mathbf{y}_d$ is a sequence of labels $\{y_{di}\}_i$, with each y_{di} indicating the label of the corresponding token. The author of each document is a user from G, and thus the social network G encodes the social context information and the labeled instances in D are mapped to concepts in KB. Thus the knowledge base provides domain knowledge.

Our goal is to learn a function from the given training dataset so as to extract from document d the set of instances $\{x_k\}$ for each concept $c_j \in C$. More specifically, we can define the problem as a sequential classification problem as follows:

$$f(\mathbf{w}_d, \theta, G, KB) \rightarrow \{(y_{di}, c_j)\}$$

where θ is the learned topic model from the input documents, y_{di} is a variable corresponding to w_{di} to represent whether w_{di} is (part of) an instance of concept c_j.

The fundamental challenge of this problem is how to capture the social context and the domain knowledge for recognizing each instance. In particular, for a specific concept/instance mentioned in someone's document (e.g., a microblog), which friends could share social context information, and on which topic?

3. PRELIMINARIES

We first introduce a baseline solution for this problem. To recognize instances from a free document, we can consider a sequential labeling model, for example, Conditional Random Fields (CRFs) [19]. Before proceeding, we first separate the document into tokens, then assign possible tags (e.g., concepts in C) to each token. The tokens form the basic units and the documents form the sequences of units in the tagging problem. Then given a training dataset, a CRF model is built with the labeled data by means of an iterative algorithm based on Maximum Likelihood Estimation, i.e.,

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z} \exp\left(\sum_i \sum_k \lambda_k f_k(x_i, y_i) + \sum_i \sum_j \mu_j f_j(\mathbf{x}, y_i, y_{i+1})\right) \quad (1)$$

where $\mathbf{x}$ denotes a sequence of the tokens, $\mathbf{y}$ is a label sequence, x_i is the i-th token in the sequence, and y_i is the label of the i-th token; $f_k(x_i, y_i)$ and $f_j(\mathbf{x}, y_i, y_{i+1})$ respectively represents the k^{th} feature function defined for individual token x_i and the j^{th} feature function defined for two consecutive tokens x_i and x_{i+1}. For example, a Boolean feature function can be defined for the token "Peace West King" and the label "politician". Finally, Z is a normalization factor to ensure that the distribution is normalized so that the sum of the probabilities equals 1. An example is illustrated in Figure 3(a).

In recognition, given a sequence of tokens $\mathbf{x}$, we determine the most likely corresponding sequence $\mathbf{y}^*$ of labels by using the trained CRF model, i.e., $\mathbf{y}^* = \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x})$.

4. SOCINST MODEL FRAMEWORK

The basic sequential labeling with CRFs only considers the linguistic information, but ignores the (topical) context information. This is very likely to result in high precision, but low recall performance (the results in § 5 confirm this). To alleviate this problem, we can resort to the topic modeling approach. However, traditional topic models cannot incorporate social context and domain knowledge information. To this end, we develop a new topic modeling called *Social Context-aware Instance Recognition (SOCINST)* to model social context and domain knowledge for entity instance recognition.

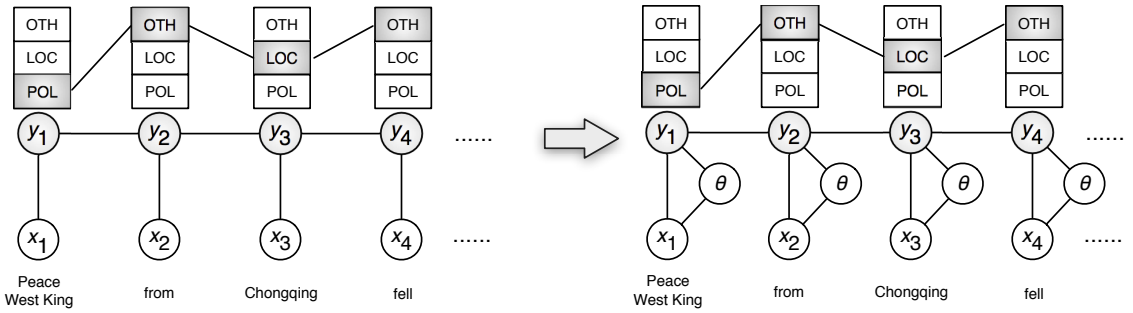
We begin with a brief introduction of the topic model and then describe the proposed method. Probabilistic topic models have been successfully applied to multiple text mining tasks to extract topics from text [5, 15, 30]. We employ an Author-Topic (AT) model [30], which utilizes the topic distribution to represent the interdependencies among authors and document content. The AT model can be considered as an extension of Latent Dirichlet Allocation (LDA) [5], but one that considers the collaborative relationships between users. The model simulates the process when people collaborate on a work, e.g., writing a scientific paper, using a series of probabilistic steps. In essence, for each object it estimates a mixture of topic distributions that represent the probability of the object associated with every topic. For example, for each author v, we have a set of probabilities $\{P(z_i|v)\}_i$ or $\{\theta_{vz_i}\}_i$, respectively denoting how likely author v is interested in topic z_i. Similarly, we have $\{P(w_j|z)\}_j$ or $\{\phi_{zw_j}\}_j$, the probability of word w_j given topic z. We use Gibbs sampling to learn the probabilities. The interested reader can refer to [30] and [28] for details. To model inter-dependencies among more categories of entities, one can also consider Author-Conference-Topic (ACT) model [35].

Based on the learned topic distributions, we define topic-based features and incorporate them into the sequential labeling method. An example is illustrated in Figure 3(b).

4.1 Model Description

The traditional topic modeling approach does not consider the social-structure information; also, it is difficult to incorporate domain knowledge into the model. As a result, the learned topical context cannot accurately represent the social context. We develop a new topic modeling approach called Social Context-aware Instance Recognition (SOCINST) to incorporate both social context and domain knowledge into the topic model. Regarding social context, the basic idea here is that if two users have strong relationships with each other, e.g., as shown by their having many interactions, then their topic distribution are likely to be similar. For incorporating domain knowledge, the idea is that if two words are instances of the same concept or two related concepts in the knowledge base, then the two words are likely to have the same topic.

Technically, the difference of the proposed model from the Author-Topic (AT) model lies in the way that θ and ϕ are generated. For modeling documents, the AT model treats each document separately, and topics in a document are sampled from the author-specific multinomial distribution θ. However, this ignores poten-



(a) Sequential Labeling with CRF (b) Sequential labeling with SOCINST

Figure 3: Graphical representation of the two sequential labeling models

tial relations between documents. Similarly, for modeling words, AT assumes that a document's words are interchangeable, and each word is sampled from a topic-specific multinomial distribution ϕ, which again ignores the potential relations between words. In the proposed SOCINST model, the multinomial distribution of instances is replaced with a tree-structured multinomial specific to the corresponding concept — e.g., $\phi \sim \text{DirichletTree}(\beta, \eta)$ — and the user-specific multinomial is replaced by a mixture of multinomials by combining neighbors' topic distributions. The advantage of this modeling method is that it smooths the learned model for instances where they belong to the same concept, which can significantly alleviate the ambiguity problem [4, 18].

Let us briefly introduce notations. d is a document and v_d is the author of the document[2]; θ_v is the topic model for author v; ϕ_z is a multinomial distribution over words specific to topic z; α and β are Dirichlet hyperparameters; η is the hyperparameter for the Dirichlet tree. Table 1 summarizes the notations used in the SOCINST model.

Modeling domain knowledge. Formally, the generative process of SOCINST is described in Algorithm 1. If a document d does not have any links to other documents or the author does not have any relationships with others, then for each word w_{di}, we draw a topic z_{di} from a topic distribution θ_v specific to author v and then use the topic to sample the word. In particular, for sampling word w_{di}, we check if the word is an instance of some concept $c \in KB$ in the training dataset. The generative process is described in Algorithm 2. If it is not, then we use a topic-specific multinomial distribution $\phi_{z_{di}}$ to generate the word. But if it is an instance of concept c, then we first sample the concept c from the multinomial $\phi_{z_{di}}$ and then sample the word from the concept-specific multinomial ψ_c. Intuitively, we replace the sampling of a word with a two-level generative process. At the first level, we select the corresponding concept according to a concept path distribution $multi(\pi)$; and at the second level, sample the word. Figure 4 shows an example of the sampling process. Theoretically, this process can be explained by the Dirichlet tree distribution [11, 31], a generalization of the Dirichlet distribution. In a Dirichlet distribution, all words in a document are treated independently; thus it is difficult to model dependencies between words. The Dirichlet tree distribution can be considered as a tree where words are leaf nodes, concepts are internal nodes, and the root node is the super concept of all nodes. Each tree edge encodes weight from the parent node to the child node. Let $S(c)$ be the immediate child of concept node c, $W(c)$ be the leaf nodes (words) in the subtree under concept c, W be all leaves

[2]For simplicity, we consider one author; but this can be easily extended to multiple authors by adding a uniform distribution for sampling an author to be responsible for each sampled topic [30].

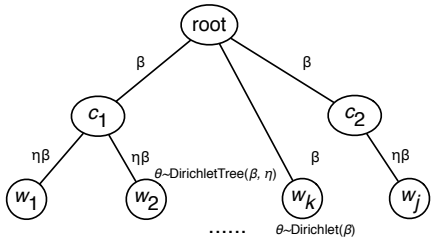


Figure 4: Example of sampling words from Dirichlet tree distribution. c is a concept and w is a word; β and η are two hyperparameters; root node is a super concept

in the tree and C be all the concepts (except the root concept). To generate a sample of multinomial distribution ϕ of a word (leaf node) from the Dirichlet tree distribution, we first sample a multinomial from the root node by using the edge weights from the root node to child nodes (concepts) as Dirichlet parameters, and further sample (re-distribute) the multinomial from the sampled internal (concept) node to its child nodes again, until we finally obtain the word. For simplicity, if we consider only one level of concepts, we can have the following Dirichlet tree distribution

$$\text{DirichletTree}(\beta, \eta) = \left(\prod_{i=1}^{W} \phi_{zw_i}^{\eta_{w_i}}\right) \times \left(\prod_{j, c_j \in C} \frac{\Gamma(\sum_{k, w_k \in W(c)} \eta_{w_k})}{\prod_{k, w_k \in W(c)} \Gamma(\eta_{w_k})} \left(\sum_{k, w_k \in W(c)} \phi_{zw_i}^{\eta_{w_i}}\right)^{\Delta(s)}\right) \quad (2)$$

where $\Gamma(.)$ is a gamma function; $\Delta(s) = \eta_c - \sum_{k, w_k \in W(c)} \eta_{w_k}$ denotes the difference between the weight of a concept and the sum of the weights of all instances under a concept. When $\Delta = 0$ for all concepts, the Dirichlet tree distribution reduces to a Dirichlet distribution.

Modeling social context. We mainly consider two types of social relationships: collaboration relationships and reference relationships. A collaboration relationship indicates that two users v_i and v_j collaborate with each other; for example collaboration on a scientific paper. A reference relationship indicates that v_i has a document that refers to one of v_j's documents. For example, in Twitter, if user v_i adds a post as a reply on v_j's microblog, then we create a reference relationship. The relationship can be either directed or undirected. For simplicity, here we consider only undirected relationships. For each relationship, (v, v'), and a new multinomial

Input: a social network G, a document set D, a knowledge base KB;
Output: estimated parameters θ,ϕ

For each author v, draw θ_v from Dirichlet prior α;
For each topic z, draw ϕ_z from Dirichlet prior β;
foreach *document* d **do**
  if v_d *does not have relationship with others* **then**
    foreach *word* $w_{di} \in \boldsymbol{w}_d$ **do**
      Draw a topic $z_{di} \sim multi(\theta_v)$ from the topic model of user v;
      Call SamplingWord(z_{di}, w_{di});
    end
  end
  else if v_d *have relationship with* v' **then**
    Construct a multinomial mixture $\vartheta_{v_d v'}$ by combining topics distributions specific to users v_d and v';
    foreach *word* $w_{di} \in \boldsymbol{w}_d$ **do**
      Draw a topic $z_{di} \sim multi(\vartheta)$ from the distribution specific to the pair;
      Call SamplingWord(z_{di}, w_{di});
    end
  end
end

Algorithm 1: Probabilistic generative process in SOCINST

`SamplingWord(`z_{di}, w_{di}`)`
if w_{di} *is an instance of a concept* $c \in KB$ **then**
  Draw a concept path $\{c_k\}_k \sim multi(\pi)$ from a topic-specific concept path distribution;
  Draw word $w_{di} \sim multi(\psi_c)$ from a concept-specific multinomial distribution;
end
else
  Draw word $w_{di} \sim multi(\phi_{z_{di}})$ directly from a topic-specific multinomial distribution;
end

Algorithm 2: SamplingWord()

distribution $\vartheta_{vv'}$ is constructed by combining the two multinomial θ_v and $\theta_{v'}$ specific to the two users v and v'. The new distribution $\vartheta_{vv'}$ is then defined as $\tau(\theta_v + \theta_{v'})$, a simple mixture of the two expanded multinomials of θ_v and $\theta_{v'}$ [7], where τ is a normalization factor to guarantee that the sum of the distribution is 1. Finally, the word w_{di} is sampled from a topic z_{di} according to the new distribution $\vartheta_{vv'}$. By modeling the new distribution, SOCINST smooths the topic distribution of users who have social relationships. A similar strategy has been also used in [34] for modeling collaborations. Figure 5 gives an example of constructing multinomial mixtures based on social relationships. It is very flexible for incorporating hierarchical structure among users. For example, two users can form a pair, and the pair can be further combined with other pairs to form a group, and further combined with other groups to form a community. To do this, we only need to add more internal nodes to the constructed Dirichlet tree.

4.2 Model Learning

The Dirichlet tree distribution is also conjugate to the multinomial, thus we can use the Markov Chain Monte Carlo (MCMC) method to effectively train the model. In particular, we use Gibbs sampling to estimate the unknown parameters $\{\theta, \vartheta, \phi\}$ in the SOCINST model. We evaluate the posterior distribution on z for each word in the document; then use the sampling results of z to infer θ, ϑ and ϕ. More specifically, we begin with the joint probability of all documents, and then using the chain rule, we obtain the posterior probability of sampling the topic for each word.

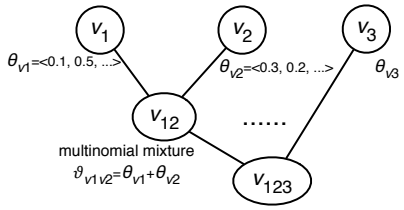


Figure 5: Example of constructing multinomial mixture based on social relationships

When we do not consider the social context, and the specific word w_{di} is not an instance of a concept in the knowledge base, we use a sampling equation similar to that in the Author-Topic model [30], i.e. with the posterior probability:

$$P(z_{di}|\mathbf{z}_{-di}, \mathbf{w}, \cdot) = \frac{n_{vz_{di}}^{-di} + \alpha}{\sum_z n_{vz}^{-di} + K\alpha} \times \frac{m_{z_{di}w_{di}}^{-di} + \beta}{\sum_w m_{z_{di}w}^{-di} + W\beta} \quad (3)$$

where n_{vz} is the number of times that topic z has been sampled from the multinomial distribution specific to the author v; m_{zw} is the number of times that word w has been generated by topic z; the number n^{-di} with the superscript $-di$ denotes a quantity, excluding the current instance; $\cdot$ indicates all the other parameters we should consider when calculating the probability.

If the word w_{di} is an instance of a concept, then based on the conjugate property between the Dirichlet tree distribution and the multinomial, we have (by assuming that we only have one level of concepts):

$$\begin{aligned} P(z_{di}|\mathbf{z}_{-di}, \mathbf{w}, \cdot) = & \frac{n_{vz_{di}}^{-di} + \alpha}{\sum_z n_{vz}^{-di} + W\alpha} \\ & \times \frac{m_{z_{di}c_{di}}^{-di} + W_c\beta}{\sum_c m_{z_{di}c}^{-di} + W\beta} \times \frac{m_{c_{di}w_{di}}^{-di} + \eta}{\sum_w m_{c_{di}w}^{-di} + W_c\eta} \end{aligned} \quad (4)$$

where m_{zc} is the number of times that instances of concept c have been generated by z; W_c is the number of instances of concept c; m_{cw} is the number of times word w appears as an instance of c in all documents. As discussed before, the model can be easily extended by incorporating more levels of concepts. For example, we could consider concept-subconcept relationships by adding an internal node into the Dirichlet tree. Accordingly, the posterior probability can be generalized as:

$$\begin{aligned} P(z_{di}|\mathbf{z}_{-di}, \mathbf{w}, \cdot) = & \frac{n_{vz_{di}}^{-di} + \alpha}{\sum_z n_{vz}^{-di} + W\alpha} \\ & \times \prod_{k=1}^{T} \frac{m_{z_{di}c_{di}^k}^{-di} + W_{c_{di}^k}\beta}{\sum_{c_s}(m_{z_{di}c_s}^{-di} + W_{c_s^k}\beta)} \times \frac{m_{c_{di}w_{di}}^{-di} + \eta}{\sum_w m_{c_{di}w}^{-di} + W_c\eta} \end{aligned} \quad (5)$$

where $\{c_{di}^1, c_{di}^2, \cdots, c_{di}^T\}$ is a concept path from the root node to the leaf word node (excluding the end nodes); c_s indicates a child node of concept c.

To incorporate the social context further into the model, we use the constructed new distribution $\vartheta_{vv'}$ to replace the original multinomial distribution of user v and v', and then use the distribution $\vartheta_{vv'}$ to sample topic for each word. The posterior probability can be rewritten as:

$$P(z_{di}|\mathbf{z}_{-di},\mathbf{w},\cdot) = \frac{n^{-di}_{vz_{di}} + \gamma n_{v'z_{di}} + \alpha}{\sum_z (n^{-di}_{vz} + \gamma n_{v'z}) + W\alpha} \times \prod_{k=1}^{T} \frac{m^{-di}_{z_{di}c^k_{di}} + W_{c^k_{di}}\beta}{\sum_{c_s}(m^{-di}_{z_{di}c_s} + W_{c^k_s}\beta)} \times \frac{m^{-di}_{c_{di}w_{di}} + \eta}{\sum_w m^{-di}_{c_{di}w} + W_c\eta} \quad (6)$$

where γ is a tunable parameter to control the extent to which we want to smooth the distribution between v and v'. (Detailed derivation is given in Appendix.)

During the parameter estimation, the algorithm keeps track of an $|V| \times K$ (user by topic) count matrix, and an $C \times K$ (concepts by topic) count matrix. Given these matrices, we can estimate the probabilities of θ, ϑ, and ϕ.

Complexity Analysis. We analyze the complexity of the proposed model. The AT model has a complexity of $O(\bar{A}_d M \bar{N}_d K L)$, where M is the number of documents, $\bar{A}_d$ is the average number of authors for document d, L is the number of sampling iterations, K is the number of topics, and $\bar{N}_d$ is the average number of word tokens in document d. In our setting, we only consider one author, thus $\bar{A}_d$ can be ignored from the complexity. Then, by incorporating the domain knowledge, the complexity increases linearly with the height T of the Dirichlet tree. The height T is usually very small compared to the scale of the knowledge base, and can be considered as a constant. By incorporating the social context, the complexity increases linearly with the average number $|\bar{E}|$ of social relationships for each user in the social network G. By combining both of them, we obtain the final complexity $O(M\bar{N}_d L|\bar{E}|K)$. The complexity is acceptable, as in most cases, the average number of social relationships $|\bar{E}|$ is small comparing with M and N_d.

4.3 Applying to Instance Recognition

We now discuss how to combine the learned topic information into sequential labeling model for instance recognition. First, we apply the proposed SOCINST model to the document set to learn the topic distributions. Then we define feature functions in the sequential labeling method based on the topic distributions for instance recognition. Specifically, we define K topic-specific feature functions in the CRF model for each token, where K is the number of topics learned by the topic model. We use a threshold τ to determine whether a topic is relevant to an instance or not. If $P(z|v) > \tau$, then the value of the corresponding unit is 1, otherwise 0. Figure 3(b) shows the graphical representation of the sequential labeling process with SOCINST.

When training SOCINST, as for the hyperparameters α, β, and η, following [1, 5], we empirically take fixed values (i.e., $K = 15$, $\alpha = 0.1$, $\beta = 0.1$, and $\eta = 10$). γ is defined to represent our preference for smoothing the distributions between users with social relationships. We set it as $\gamma = 0.5$. We did try different settings and found that the estimated topic models are not very sensitive to the hyperparameters.

5. EXPERIMENTAL RESULTS

We conduct various experiments to evaluate the SOCINST method. All datasets and codes are publicly available.[3]

5.1 Experiment Setup

Dataset. We evaluate the proposed method on three datasets (as shown in Table 2): ICDM'12 Contest, Weibo, and I2B2.

[3]http://aminer.org/socinst/

Table 2: Statistics of the three datasets

Dataset	Weibo	I2B2	ICDM'12 Contest
#documents	1,800	899	2,110
#instances	545	2,400	565
#relationships	10,763	27,175	NA

Weibo. Weibo is the most popular microblogging service in China. The dataset is from [17]. The dataset includes 1,553,347 tweets crawled from Weibo from May 1st, 2013 to June 30th, 2013. Human annotations have been made on the dataset to label morph entities (e.g., "Fruit company") and their corresponding targets ("Apple Inc."). In total, there are 107 different morph entities. We view each distinct morph entity as a concept in the knowledge base. We randomly sampled 1,800 tweets that contain the morph entities. We use the following relationship to construct the relationship between users. Finally, we extracted 10,763 relationships. Our goal is to extract real morph instances in the dataset.

I2B2. This is a health care dataset from [37]. It was used in the 2006 Deidentification Challenge on automatically identifying private health information from medical discharge records.[4] The dataset comprises 899 medical records, with a total of 2,400 private health information instances in the records. We view each patient as a user and create a relationship between patients if they go to the same hospital. In total, we have 27,175 relationships. The knowledge base consists of eight concepts, such as Doctor, Location, and Hospital. Our goal here is to extract private health information instances in the dataset.

ICDM'12 Contest. This is a product forum dataset used in the ICDM'12 Contest[5]. The task is to automatically recognize product mentions in forum textual content and to also disambiguate which product(s) in product catalogs are being referenced. The dataset comprises 2,110 documents and a total of 565 labeled product mentions in these documents. We view each product catalog as a concept in the knowledge base and finally construct a large knowledge base of 15,367,328 concepts. The dataset does not provide the user information; thus, we mainly focus on evaluating the effect of incorporating domain knowledge for instance recognition. Our goal is to recognize product mentions in the dataset.

Comparison methods. We compare our model with the following methods for instance extraction:

Standard Matching (SM): Simply extracts all the terms/symbols that are annotated as extracted instances in the training data, then finds their occurrences in the test data and extracts them as target instances.

Rule Template (RT): Recognizes target instances from the test data by a set of rule templates. We design rules based on sentence position, special characters, and semantic patterns to extract instances.

Conditional Random Field (CRF): Trains a conditional random field (CRF) model using features associated with each token. Classifies each token into predefined labels, such as age, location, and phone in the I2B2 dataset. For CRF, we employ MALLET [24].

CRF+AT: Uses Author-Topic (AT) [30] to train a model for each document. Then it incorporates the topic distribution of each document as features in the CRF model for instance recognition. The

[4]https://www.i2b2.org/NLP/DataSets/Main.php

[5]http://icdm2012.ua.ac.be/content/contest

Table 3: Performance of different methods on the three datasets (%)

Data	Method	Recall	Precision	F1-Measure
Weibo	SM	55.34	34.92	42.82
	RT	39.62	66.31	49.60
	CRF	29.24	94.89	44.71
	CRF+AT	43.71	89.67	58.77
	SOCINST	**65.72**	76.27	**70.60**
I2B2	SM	39.58	28.24	32.96
	RT	39.60	40.29	39.94
	CRF	40.99	56.19	47.40
	CRF+AT	41.37	54.92	47.19
	SOCINST	**43.94**	**57.18**	**49.69**
ICDM'12 Contest	SM	9.47	62.50	16.46
	RT	23.69	42.01	30.30
	CRF	21.80	53.48	30.97
	CRF+AT	26.54	51.37	35.00
	SOCINST	**37.91**	53.33	**44.32**

difference between this method and CRF is that here we consider topic information.[6]

SOCINST: (Cf. § 4) This is the proposed method, which incorporates both domain knowledge and social context to extract topics. The topical information is then integrated into the sequential labeling method for instance recognition.

SM and RT only consider hard rules. In all the other comparison methods, we try to use the same linguistic features. Unlike CRF, CRF+AT and SOCINST incorporate topic-based features. The difference between AT and SOCINST lies in the way that topics are learned from the documents.

Evaluation Measures. In each dataset, we use the different methods to recognize instances and compare with ground-truth data. We evaluate the performance of different approaches in terms of Precision, Recall, and F1-Measure [6].

All algorithms are implemented in C++ and Python, and the experiments are performed on an x64 machine with E5-4650 2.70GHz Intel Xeon CPU (with 64 cores) and 128GB RAM. The operating system is Ubuntu 12.04. The proposed algorithm has tractable running times on the datasets and requires less than 120 minutes for training and prediction.

5.2 Performance Analysis

Table 3 lists the performance of instance recognition by the comparison methods on the three datasets. The proposed SOCINST method clearly outperforms the comparison methods (+5.3-26.6% in terms of F1-score, $p < 0.01$ with t-test). SM and RT use hard rules for recognizing instances, which often leads to suboptimal performance in F1-measure. CRF, considering the statistical linguistic information, improves recognition performance in terms of F1-measure. AT incorporates the topic information extracted from the input documents, and thus performs better than the standard CRF method. SOCINST incorporates both domain knowledge and social context into the topic model and obtains significant improvement over both CRF and CRF+AT methods. On the ICDM'12 Contest dataset, we further compare SOCINST with the method [38] of the first place in the contest. In the comparison, we use the same *blending strategy* as that of [38] to combine results of different models. Figure 6 shows a performance comparison of the two methods. SOCINST Blend is the result of our method with the blending strategy. It can be seen that our method significantly outperforms (+21.4%; $p \ll 1e-5$ with t-test) the performance of the first place in terms of all measures.

[6]The method is similar to T-NER [27], which considers both labeled information and topic information for instance recognition, but it does not model social context. We compare with this method to demonstrate the necessary to model social context and domain knowledge together.

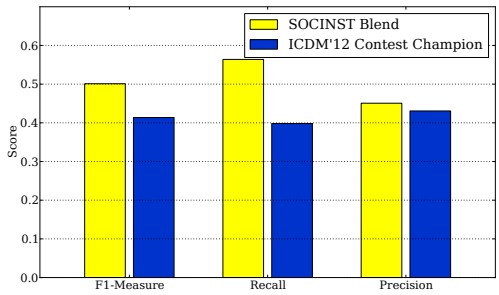


Figure 6: Performance comparison of SOCINST and the first place [38] in ICDM'12 Contest

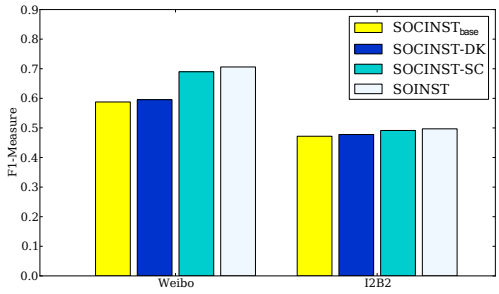
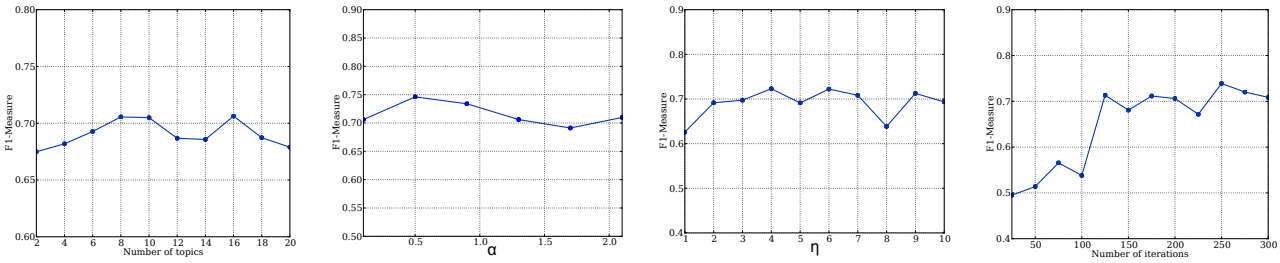


Figure 8: Effects of social context and domain knowledge. For SOCINST$_{base}$, we removed both social context and domain knowledge from our method. SOCINST-SC and SOCINST-DK are results in which we removed social context or domain knowledge information, respectively, from our method

Effects of social context and domain knowledge. We study how social context and domain knowledge can help instance recognition. Since the ICDM'12 Contest dataset does not consist of social relationships, we focus this analysis on the other two datasets: Weibo and I2B2. More specifically, we respectively remove domain knowledge and social context when training our proposed model, and then compare performance of instance recognition based on the trained models. Figure 8 shows the F1-Measure performance on the two datasets. SOCINST$_{base}$ means that we removed both social context and domain knowledge from our method. SOCINST-SC and SOCINST-DK indicate results from having removed social context or domain-knowledge information, respectively, from our method. We can clearly see that both social context and domain knowledge contribute significantly to the results for the two datasets. It is also interesting to see that social context seems to be more important for modeling the Weibo data. With social context information, the performance of instance recognition improved by up to 10% on Weibo, compared to an improvement of 5% on the I2B2 data.

Parameter analysis. We evaluate how different parameters (including number of topics, K, and hyperparameters α and η) affect



(a) number of topics K (b) Hyperparameter α (c) Hyperparameter η (d) Convergence analysis

Figure 7: Parameter analysis. (a) Performance of the SOCINST model by varying the number of topics K; (b) Performance of the SOCINST model is stable when varying α parameter; (c) Performance of SOCINST when varying the parameter η; (d) Convergence analysis of SOCINST model

the quality of the models learned by SOCINST. We performed the following analysis based on the Weibo data. For the number of topics, K, we perform an analysis by varying the number of topics in the proposed SOCINST method. Figure 7(a) shows its F1-Measure performance with the number of topics varied. We see that when the number is small (< 10), increasing the number often results in a performance improvement. The trend becomes stable when the number of topics is about 10. This demonstrates the stability of the SOCINST method with respect to the number of topics. Regarding the hyperparameter α, Figure 7(b) shows the performance of SOCINST with the parameter α varied (all the other hyperparameters fixed and the number of topics is set to $K = 15$). Although the performance changes when varying the value of α, the largest difference is less than 0.03. This confirms that the SOCINST method is not sensitive to the particular choice of α. Regarding the parameter η, Figure 7(c) shows the performance of SOCINST with the parameter η varied (with all the other hyperparameters fixed). It indicates that for a very small value for η, which means that we largely ignore the effect of social context, the performance varies quite a bit. However, as we increase η to 6 or more, the performance becomes much more stable.

Convergence analysis. We further investigate the convergence of the SOCINST learning algorithm. Figure 7(d) shows the convergence analysis of the algorithm on the Weibo dataset. We see that the algorithm converges within 300 Gibbs sampling iterations. This rapid fast convergence makes possible efficient training of the model on large scale datasets.

5.3 Discussions

Leveraging social context and domain knowledge, SOCINST clearly outperforms traditional instance recognition methods, such as Standard Matching (SM), Rule Template (RT), and Conditional Random Field (CRF). Several recently developed methods also considered using topic model to improve the performance of entity recognition. T-NER [27] is one of the most closely related methods. It considers labeled information and topic information for instance recognition. T-NER performs better than the Stanford NER system. In our comparison methods, CRF+AT can be considered as a counterpart of T-NER. CRF+AT considers labeled information, as well as topic information. However both T-NER and CRF+AT do not consider social context. CRF+AT underperforms the proposed SOCINST method by -2.5-11.9%. Liu et al. [23] presented another related method that incorporates additional (redundancy) information into a linear Conditional Random Fields (CRF) model for entity recognition. In principle, this method is similar to CRF or CRF+AT in our comparison methods, because the only difference between the method and CRF is that it combines the K-Nearest Neighbors (KNN) classified results into the CRF model to boost the recognition performance.

Based on the propose model, we are developing a new feature in ArnetMiner [35][7], an academic social network analysis and mining system. We are trying to automatically recognize instances from paper abstracts (or user queries) and map the the instances to a knowledge base from Wikipedia.

6. RELATED WORK

Considerable research has been conducted on entity recognition. Collins [9] proposed a ranking algorithm for entity extraction based on boosting and the voted perceptron. The method can obtain a performance similar to that of the Conditional Random Field (CRF) method. Finkel et al. [13] designed an entity extraction method by combining long-distance dependency information. A survey of entity recognition can be also found in [25]. However, all these works mainly focus on the linguistic information, and do not consider the social context or domain knowledge. Recently, Liu et al. [23] proposed to combine a K-Nearest Neighbors (KNN) classifier with a linear CRF model under a semi-supervised framework to deal with the unavailability of training data. This method can be considered as an extension of the CRF method, but it does not explicitly consider social context and domain knowledge information. Ritter et al. [27] used Labeled LDA [26] to exploit Freebase as a source of distant supervision to help named entity recognition in tweets by leveraging the redundancy in tweets. Again, they do not consider the social network information. Huang et al. [17] studied a special problem of entity recognition, entity morph. They try to identify those instances where authors deliberately hide the true entities due to Internet censorship. They defined several social-based features. They mainly focus on a special case of instance recognition and their method does not incorporate social context and domain knowledge in a unified model. Some other related references can be also found in [8, 20].

Our work is also relevant to entity resolution, where the task is to link entities of the same meaning or distinguish different entities with the same name. For example, Bhattacharya and Getoor [4] proposed a collective method for entity resolution in relational data. Kataria et al. [18] developed a hierarchical topic model for resolving different entities that have the same name. Li et al. [22] used network structure for named entity resolution. Tang et al.[32] studied the name resolution problem in digital libraries. However, in most entity resolution research, entity instances are assumed as input, so their focus is very different from the instance recognition studied in this work. Another line of loosely related research is en-

[7] http://aminer.org

tity matching [2, 3, 21, 29], which tries to find alignment between entities from different sources.

From the modeling aspect, substantial research has been conducted for topic models, such as [5, 15, 30]. Andrzejewski et al. [1] proposed a new method that incorporates must-links and cannot-links into the topic model. A must-link means that two words must be sampled from the same topic and a cannot-link means two words cannot be sampled from the same topic. Hu et al. [16] adopted the same strategy for modeling linguistic constraints in the topic model. In this paper, we propose a new method by combining social relationships and domain knowledge in a unified model.

7. CONCLUSION

In this paper, we study the problem of instance recognition by incorporating social context and domain knowledge. We precisely define the problem and propose a topic modeling approach to learn topics by considering social relationships between users and context information from a domain knowledge base. Experimental results on three different datasets validate the effectiveness and the efficiency of the proposed method.

The general idea in this paper, to incorporate social context and domain knowledge for entity instance recognition, represents an interesting and new research direction. There are many potential future directions for this work. A straightforward task would be to incorporate human feedback into the proposed model. Looking further ahead, we believe that combining the sequential labeling model and the proposed SOCINST into a unified model should be beneficial. Finally, further incorporating other social interactions, such as social influence [33], to help instance recognition is an intriguing direction for future research.

Acknowledgements. The work is supported by the National High-tech R&D Program (No. 2014AA015103), National Basic Research Program of China (No. 2014CB340506), Natural Science Foundation of China (No. 61222212), a research fund supported by Huawei Inc., and Beijing key lab of networked multimedia.

8. REFERENCES

[1] D. Andrzejewski, X. Zhu, and M. Craven. Incorporating domain knowledge into topic modeling via dirichlet forest priors. In *ICML'09*, pages 25–32, 2009.

[2] X. Bai, F. P. Junqueira, and S. H. Sengamedu. Exploiting user clicks for automatic seed set generation for entity matching. In *KDD'13*, pages 980–988, 2013.

[3] K. Bellare, S. Iyengar, A. G. Parameswaran, and V. Rastogi. Active sampling for entity matching. In *KDD'12*, pages 1131–1139, 2012.

[4] I. Bhattacharya and L. Getoor. Collective entity resolution in relational data. *ACM Transactions on Knowledge Discovery from Data*, 1(1):1–36, March 2007.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[6] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *SIGIR'2004*, pages 25–32, 2004.

[7] W. Buntine and A. Jakulin. Applying discrete pca in data analysis. In *UAI'04*, pages 59–66, 2004.

[8] L. Chiticariu, R. Krishnamurthy, Y. Li, F. Reiss, and S. Vaithyanathan. Domain adaptation of rule-based annotators for named-entity recognition tasks. In *EMNLP'10*, pages 1002–1012, 2010.

[9] M. Collins. Ranking algorithms for named-entity extraction: boosting and the voted perceptron. In *ACL'02*, pages 489–496, 2002.

[10] M. Dean, G. Schreiber, S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein. Owl web ontology language reference. w3c recommendation., Feb. 2004.

[11] S. Y. Dennis. On the hyper-dirichlet type 1 and hyper-liouville distributions. *Communications in Statistics - Theory and Methods*, 20:4069–4081, 1991.

[12] A. Doucet, N. de Freitas, K. Murphy, and S. Russell. Rao-blackwellised particle filtering for dynamic bayesian networks. In *UAI'00*, pages 176–183, 2000.

[13] J. R. Finkel, T. Grenager, and C. D. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL'05*, pages 363–370, 2005.

[14] G. Heinrich. Parameter estimation for text analysis. Technical report, University of Leipzig, Germany, 2004.

[15] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR'99*, pages 50–57, 1999.

[16] Y. Hu, J. Boyd-Graber, and B. Satinoff. Interactive topic modeling. In *HLT'11*, pages 248–257, 2011.

[17] H. Huang, Z. Wen, D. Yu, H. Ji, Y. Sun, J. Han, and H. Li. Resolving entity morphs in censored data. In *ACL'13*, pages 1083–1093, 2013.

[18] S. Kataria, K. S. Kumar, R. Rastogi, P. Sen, and S. H. Sengamedu. Entity disambiguation with hierarchical topic models. In *KDD'11*, pages 1037–1045, 2011.

[19] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML'01*, pages 282–289, 2001.

[20] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee. Twiner: Named entity recognition in targeted twitter stream. In *SIGIR'12*, pages 721–730, 2012.

[21] J. Li, J. Tang, Y. Li, and Q. Luo. Rimom: A dynamic multi-strategy ontology alignment framework. *IEEE TKDE*, 21(8):1218–1232, 2009.

[22] Y. Li, C. Wang, F. Han, J. Han, D. Roth, and X. Yan. Mining evidences for named entity disambiguation. In *KDD'13*, pages 1070–1078, 2013.

[23] X. Liu, S. Zhang, F. Wei, and M. Zhou. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ACL '11, pages 359–367, 2011.

[24] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[25] D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30:3–26, 2007.

[26] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP '09*, pages 248–256, 2009.

[27] A. Ritter, S. Clark, Mausam, and O. Etzioni. Named entity recognition in tweets: An experimental study. In *EMNLP'11*, pages 1524–1534, 2011.

[28] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *UAI'04*, pages 487–494, 2004.

[29] W. Shen, J. Wang, P. Luo, and M. Wang. Linking named entities in tweets with knowledge base via user interest modeling. In *KDD'13*, pages 68–76, 2013.

[30] M. Steyvers, P. Smyth, and T. Griffiths. Probabilistic author-topic models for information discovery. In *KDD'04*, pages 306–315, 2004.

[31] Y.-C. Tam and T. Schultz. Correlated latent semantic model for unsupervised lm adaptation. In *ICASSP?07*, volume 4, pages IV–41–IV–44, 2007.

[32] J. Tang, A. Fong, B. Wang, and J. Zhang. A unified probabilistic framework for name disambiguation in digital library. *IEEE TKDE*, 24(6):975–987, 2012.

[33] J. Tang, J. Sun, C. Wang, and Z. Yang. Social influence analysis in large-scale networks. In *KDD'09*, pages 807–816, 2009.

[34] J. Tang, S. Wu, J. Sun, and H. Su. Cross-domain collaboration recommendation. In *KDD'12*, pages 1285–1294, 2012.

[35] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: Extraction and mining of academic social networks. In *KDD'08*, pages 990–998, 2008.

[36] K. M. Ting and I. H. Witten. Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289, 1999.

[37] O. Uzuner, Y. Juo, and P. Szolovits. Evaluating the state-of-the-art in automatic de-identification. *J Am Med Inform Assoc*, 14(5):550–563, 2007.

[38] S. Wu, Z. Fang, and J. Tang. Accurate product name recognition from user generated content. In *ICDM 2012 Contest*, pages 874–877, 2012.

9. APPENDIX

According to the generative process, we could integrate out the multinomial distributions θ, ϑ, ϕ, because the model only uses conjugate priors [12]. We use Eq. 6 as the example to explain its derivation, as it contains both social context and domain knowledge. First we write the joint probability:

$$\begin{aligned} &P(\mathbf{w},\mathbf{z},\mathbf{v}|\alpha,\beta,\eta,\gamma) \\ \propto &\int P(\mathbf{z}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta \\ &\int P(\mathbf{w}|\mathbf{z},\phi)P(\phi|\pi_T)\prod_{k=1}^{T}P(\pi_{k+1}|\pi_k,\eta_c)P(\pi_1|\beta,\eta)d\phi d\pi \end{aligned} \tag{7}$$

The conditional of s_i is obtained by dividing the joint distribution of all variables by the joint with all variables but s_i (denoted by $\mathbf{s}_{-i}$) and canceling factors that do not depend on $\mathbf{s}_{-i}$.

$$\begin{aligned} &P(z_{di}|\mathbf{z}_{-di},\mathbf{w},.) = \frac{P(\mathbf{w},\mathbf{z},\mathbf{v}|\alpha,\beta,\eta,\gamma)}{P(\mathbf{w},\mathbf{z}_{di},\mathbf{v}|\alpha,\beta,\eta,\gamma)} \\ =&\frac{\int P(\mathbf{w}|\mathbf{z},\phi)P(\phi|\pi_T)\prod_{k=1}^{T}P(\pi_{k+1}|\pi_k,\eta_c)P(\pi_1|\beta,\eta)d\phi d\pi}{\int P(\mathbf{w}|\mathbf{z},\phi)P(\phi|\pi_T)\prod_{k=1}^{T}P(\pi_{k+1}|\pi_k,\eta_c)P(\pi_1|\beta,\eta)d\phi d\pi} \\ &\cdot\frac{\int P(\mathbf{z}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta}{\int P(\mathbf{z}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta} \end{aligned} \tag{8}$$

The first fraction of Eq. 8 is responsible for sampling word from topic and the second term is responsible for sampling topic from user (with social relationships). We start with the derivation of the second fraction. Specifically, as $P(\mathbf{z}|(\mathbf{v},\mathbf{v}',\vartheta)$ and $P(\vartheta|\alpha)$ are a conjugate pair of Multinomial-Dirichlet, we could solve the Multinomial-Dirichlet integral using Gibbs sampling [14]:

$$\begin{aligned} &\int P(\mathbf{z}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta \\ =&\prod_d\frac{1}{\Delta(\alpha)}\prod_z\vartheta_{vv'z}^{n_{vz}+n_{v'z}+\alpha-1}d\vartheta_{vv'} \\ =&\prod_d\frac{\Delta(\vec{n_d}+\alpha)}{\Delta(\alpha)}, \\ &\text{with } \Delta(\alpha)=\frac{\Gamma(\alpha)^T}{\Gamma(T\alpha)} \text{ and } \vec{n_d}=\{n_{vz}+n_{v'z}\}_{z=1}^{T} \end{aligned}$$

where (v,v') denotes a social relationship between user v and v'; n_{vz} and $n_{v'z}$ are two numbers obtained when combining the two distributions θ_v and $\theta_{v'}$. Essentially, in this sampling, we smooth the sampled topic from a user v-specific topic distribution θ_v by the mixture $\vartheta_{vv'}$ of θ_v and $\theta_{v'}$. Accordingly, the second fraction of Eq. 8 can be written as:

$$\begin{aligned} &\frac{\int P(\mathbf{z}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta}{\int P(\mathbf{z}_{-di}|(\mathbf{v},\mathbf{v}'),\vartheta)P(\vartheta|\alpha)d\vartheta}=\frac{\prod_d\frac{\Delta(\vec{n_d}+\alpha)}{\Delta(\alpha)}}{\prod_d\frac{\Delta(\vec{n}_{d,\neg i}+\alpha)}{\Delta(\alpha)}} \\ =&\frac{\frac{\Gamma(n_{vz}^{-di}+n_{v'z}+\alpha)}{\Gamma(\sum_{z'}(n_{vz'}^{-di}+n_{v'z'}+\alpha))}}{\frac{\Gamma(n_{vz}^{-di}+n_{v'z}+\alpha-1)}{\Gamma([\sum_{z'}(n_{vz'}^{-di}+n_{v'z'}+\alpha)]-1)}}=\frac{n_{vz_{di}}^{-di}+n_{v'z_{di}}+\alpha}{\sum_z(n_{vz_{di}}^{-di}+n_{v'z_{di}}+\alpha)} \end{aligned} \tag{9}$$

Here, we use the identity $\Gamma(x+1)=x\Gamma(x)$; the superscript $^{-di}$ denotes a quantity, excluding the current instance. By further considering a tunable parameter γ to balance the importance between n_{vz} and $n_{v'z}$, we can obtain the first term in Eq. 6. Analogously, we can derive the first fraction of Eq. 8. The difference is that ϕ is sampled from a Dirichlet tree distribution instead of a Dirichlet distribution as that used for sampling topic z. To make it more clear, let us assume that there is a concept path $\{c_{di}^1,\cdots,c_{di}^T\}$ from the root node to the leaf word node (excluding the leaf node). To sample the first level concept c^1, we have

$$\begin{aligned} &P(c_{di}^1|\pi,\beta,\eta)=\frac{\prod_i\frac{\Delta(\vec{m}_z+\alpha)}{\Delta(\beta)}}{\prod_i\frac{\Delta(\vec{m}_{z,\neg di}+\alpha)}{\Delta(\beta)}} \\ =&\frac{\frac{\Gamma(m_{zc_{di}^1}^{-di}+W_{c_{di}^1}\beta)}{\Gamma(\sum_{c_s}(m_{zc_s}^{-di}+W_{c_s}\beta))}}{\frac{\Gamma(m_{zc_{di}^1}^{-di}+\beta-1)}{\Gamma([\sum_{c_s}(m_{zc_s}^{-di}+W_{c_s}\beta)]-1)}}=\frac{m_{zc_{di}^1}^{-di}+W_{c_{di}^1}\beta}{\sum_{c_s}(m_{zc_s}+W_{c_s}\beta)} \end{aligned} \tag{10}$$

We continue to sample the child node in the concept path, until we get the last internal node in the Dirichlet tree. In this way, we could obtain

$$\prod_{k=1}^{T}\frac{m_{z_{di}c_{di}^k}^{-di}+W_{c_{di}^k}\beta}{\sum_{c_s}(m_{z_{di}c_s}^{-di}+W_{c_s^k}\beta)} \tag{11}$$

Then applying a similar sampling process for word from a topic-specific distribution to that in the standard LDA model, we can obtain the first term in Eq. 6.

$$\frac{m_{c_{di}w_{di}}^{-di}+\eta}{\sum_w m_{c_{di}w}^{-di}+W_c\eta} \tag{12}$$

Finally, by combining Eqs. 9, 11, and 12, we obtain Eq. 6.