Entity Matching across Heterogeneous Sources

Yang Yang[†], Yizhou Sun[‡], Jie Tang[†♯⋆], Bo Ma[♭], and Juanzi Li[†]
[†]Department of Computer Science and Technology, Tsinghua University
[♯]Tsinghua National Laboratory for Information Science and Technology (TNList)
[‡]Department of Computer Science, Northeastern University
[♭]Department of Computer Science, Carnegie Mellon University
{sherlockbourne, mabodx}@gmail.com, {jietang, lijuanzi}@tsinghua.edu.cn, yzsun@cs.neu.edu

ABSTRACT

Given an entity in a source domain, finding its matched entities from another (target) domain is an important task in many applications. Traditionally, the problem was usually addressed by first extracting major keywords corresponding to the source entity and then query relevant entities from the target domain using those keywords. However, the method would inevitably fails if the two domains have *less or no overlapping* in the content. An extreme case is that the source domain is in English and the target domain is in Chinese.

In this paper, we formalize the problem as entity matching across heterogeneous sources and propose a probabilistic topic model to solve the problem. The model integrates the topic extraction and entity matching, two core subtasks for dealing with the problem, into a unified model. Specifically, for handling the text disjointing problem, we use a cross-sampling process in our model to extract topics with terms coming from all the sources, and leverage existing matching relations through latent topic layers instead of at text layers. Benefit from the proposed model, we can not only find the matched documents for a query entity, but also explain why these documents are related by showing the common topics they share. Our experiments in two real-world applications show that the proposed model can extensively improve the matching performance (+19.8% and +7.1% in two applications respectively) compared with several alternative methods.

Categories and Subject Descriptors

H.2.8 [**Database Applications**]: Data Mining; H.3.3 [**Information Search and Retrieval**]: Retrieval Models

Keywords

Heterogeneous sources; Cross-lingual matching; Topic model

1. INTRODUCTION

With the rapid growth of the Web, including online digital libraries, online social and information networks, and E-commerce systems, the Web provides abundant information to describe entities from different sources. Given an entity in a source domain, finding its matched entities from another (target) domain is an important task in many applications. For example, a patent expert may be interested in finding related patents in a patent database for a product; a user may be interested in finding all the related Chinese Wiki pages for a particular English Wiki page; and a doctor may be interested in finding all related drugs for a specific disease. Similar search problems can be found in many other applications.

The problem can be generalized as an entity matching problem across corpora from heterogeneous sources. In other words, given an entity (e.g., product) in one source, the goal is to find related entities (e.g., patents) from a different source. Despite many studies on entity matching tasks [23, 22, 3, 13, 23]. Different from traditional search tasks, one key challenge of such problem is that different sources of corpora may use rather different languages or terminologies even when describing the same topic. For example, the terms used to express the same topic about Siri, are quite different in Wikipedia and patents. As Figure 1 (a) shows, the Siri Wiki article uses more daily expressions (e.g., "voice control," "personal assistant," "iPhone," etc.) to describe Siri, in order to make it easier to understand by everyone. However, more professional and technical terms are used in patents (e.g., "information retrieval," "heuristic modules," "computer-readable medium," etc.). The descriptions of two related entities from different sources can be very dissimilar in terms of their text similarity, and thus the traditional text-based search can no longer solve the problem. In addition, for each relevant entity, it would be interesting to know on which topic the target entity is relevant to the source entity. For example, as shown in Figure 1 (a), the patent "Method for improving voice recognition" is talking about "voice control" and its relevance probability to the source Wiki article on this topic is 0.83, while the relevance probability of the second patent is 0.54 but on topic "ranking".

One possible solution is to map two entities into the same latent topic space. Intuitively, two entities are relevant to each other if they refer to the same topic, e.g., a Wiki article and a patent article should be relevant if they are both talking about the topic of Siri. A topic in such case should contain terms from heterogeneous sources. For example, the topic of Siri should contain both the general terms in Wiki and the special terms in the related patents. If we can extract hidden topics from heterogeneous sources, we will be able to infer the relevance score between two entities. However, for most topic modeling methods, such as PLSA [11] and LDA [4], they do not deal with the issue of heterogeneous sources and are not able to generate topics with terms from different sources, since these terms seldom appear in the same entities.

In this paper, we propose a novel probabilistic model, Cross-Source Topic (CST) model, to solve the entity matching problem

KDD'15, August 10-13, 2015, Sydney, NSW, Australia.

DOI: http://dx.doi.org/10.1145/2783258.2783353.

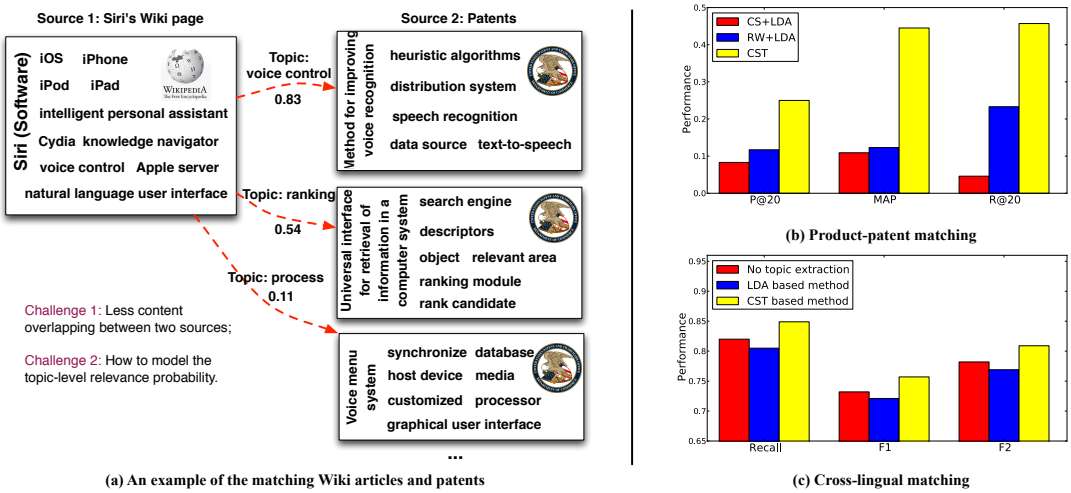


Figure 1: (a) An example of the entity between Wiki articles and patents. The rectangle on the left side represents the Wiki article which gives a general description of Siri. The rectangles on the right side denote patents reporting related technologies to Siri. Titles and high frequency phrases in the entities are shown in the rectangles. Links between the Wiki article and patents indicate their matching relations, with the topic relevance probability presented. (b) Product-patent matching performance of LDA based methods and the proposed model (CST). (c) Cross-lingual matching performance of a method not considering latent topics, a LDA based method, and a CST based method.

for a two-source case, which integrates the topic extraction and entity matching into a unified model. We first ask the users to give a small portion of labels indicating the matching between entities from heterogeneous sources. Then we model both the hidden topics and the entity matching in a unified framework, where a topic contains terms from heterogeneous sources and the entity matching is determined by the topic distributions of the two entities. By using this model, we can not only find the matched entities for a query entity, but also explain why these entities are related by showing the common topics they share. It turns out that our model can successfully overcome the little-text-overlap problem across heterogeneous corpus sources, by modeling a topic with terms coming from all the sources and utilizing the matching labels for entities across different sources. A mean-field variational inference [28, 12] method is used to learn the model, which can be used to infer the matching relation between entities with no labels.

We evaluate the CST model in two real scenarios: 1) given a Wiki article describing a specific product, searching patents in the online patent database USPTO[1] that are related to the same product; 2) given an English Wikipedia article, searching the corresponding article from the Chinese Wiki knowledge base. Figure 1 (b)-(c) show the experimental results in each scenario respectively, from which we can see that the proposed model extensively improve the performance (averagely +19.8% and +7.1% in two real scenarios respectively).

In all, our contributions of this paper are summarized in the following.

- We identify and formalize a new problem called *entity matching across heterogeneous sources*, which is important and useful in this age of plentiful online open sources from different domains. To the best of our knowledge, no previous work has extensively studied this problem.
- We propose a novel and powerful probabilistic model, Cross-Source Topic (CST) model, to solve the entity matching problem for a two-source case, which integrates the topic extraction and matching into a unified model.
- We design an efficient variational inference-based learning algorithm to learn the model and enable it scale to large-scale data sets.
- We have demonstrated the power of our new method using two real-world applications, compared with the state-of-the-art baselines.

Organization Section 2 formulates the problem. Section 3 explains our proposed model, describes the algorithm for model learning, and introduces the applications of the model. Section 4 introduces our experiment that validates the effectiveness of our methodology, including its setup, baseline methods and results. Section 5 reviews some related work, and finally, Section 6 concludes this work.

2. PROBLEM DEFINITION

In this section, we present related definitions and formulate the problem. We first give the formal definition of heterogeneous source corpus. Generally, a heterogeneous source corpus contains the descriptions of entities from multiple sources. However, to make the definition and the description of the proposed model clear, we use a dual source corpus as an instance in all related definitions. We leave the source extension as future work.

DEFINITION 1. ***Dual Source Corpus.*** *A dual source corpus C is a set of text collections $\{C_1, C_2\}$ from two sources with vocabulary $V_t = \{w_1^t, w_2^t, ..., w_{N_t}^t\}$ ($t \in \{1, 2\}$), where $C_t = \{d_1^t, d_2^t, ..., d_{D_t}^t\}$ is a collection of entities (each entity is represented by a document describing it) from source t, D_t is the number of entities in C_t, and N_t is the total number of words in*

[1]http://www.uspto.gov/

V_t. *Following the common assumption of bag-of-words representation, each entity d_i^t in C_t can be represented as a bag of words $\{w_{i_1}^t, w_{i_2}^t, ..., w_{i_{N_i^t}}^t\}$, where N_i^t is the number of words in the entity d_i^t.*

Given a dual source corpus, we can extract cross-source topics, which contain terms from different sources:

DEFINITION 2. ***Cross-Source Topic.*** *A cross-source topic φ contains multiple multinomial distributions over words from different sources. For example, a 2-source topic contains two word distributions $P_1(w|\varphi)$ and $P_2(w|\varphi)$, where $P_t(w|\varphi)$ defines the probability of a word w from source t ($t \in \{1, 2\}$) appearing in this topic. Thus words with highest probabilities associated with each topic would suggest the semantics represented by the topic. Notice that we have $\sum_{w \in V_t} p_t(w|\varphi) = 1$ ($t \in \{1, 2\}$) for any cross-source topic φ.*

Next, we use a matching relation matrix to represent the correlations between entities from different sources.

DEFINITION 3. ***Matching Relation Matrix.*** *A matching relation matrix L represents the matching status between entities in a dual source corpus C. If d_i^1 and d_j^2 is matched, $l_{i,j} = 1$, otherwise $l_{i,j} = -1$. $l_{i,j} =?$ denotes that the value is missing and needs to be inferred.*

Since entities from different sources may share few terms, the known values in the matching relation matrix are important guidance to extract the cross-source topics and infer the missing values in the matrix. We can finally define the main problem addressed in this paper:

PROBLEM 1. ***Entity Matching across Heterogeneous Sources.*** *Given a heterogeneous source corpus C, and a matching relation matrix L. The goal of cross-source entity matching is to determine the missing values in L.*

For example, we have a dual source corpus from Wikipedia and USPTO, the cross-source entity matching problem is: given a Wiki article describing a specific product, finding patents from USPTO which report the technologies related with the product. As an example, given a Wiki article describing Siri, one of the matched patent could be the one claims on the technology about "universal interface for retrieval of information," which is highly relevant to Siri.

Another example is cross-lingual Wiki article matching. Given an English Wiki article, the task aims to find a Chinese Wiki article that reports the same content. Compared with cross-lingual information retrieval problems, which mostly incorporate bilingual dictionaries, however, our problem is more general. Instead of using dictionaries, we focus on utilizing known relations to help extract cross-source topics and infer unknown relations.

3. CROSS-SOURCE TOPIC MODEL

Modeling cross-source matching entities is a challenging task. Intuitively, two entities are relevant if they refer to the same topic, and topic extraction will help us infer the connection between entities. However, due to the different terminologies used in different domains, word distributions of corpora from two sources may be quite different. In this situation, traditional topic modeling technologies would fail to identify the same topic from two sources but separate the topic into two or more, as shown in our Siri example in Figure 1. In this paper, we propose a new semi-supervised probabilistic model called Cross-Source Topic (CST) model to capture the cross-source topics and perform entity matching from different sources simultaneously.

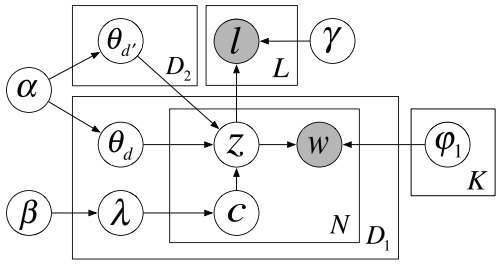


Figure 2: Plate representation of the Cross-Source Topic model. Modeling part for entities in source 2 has a symmetrical structure as source 1. For simplicity, the modeling part for the entities in source 2 is omitted.

Table 1: Notations in the CST model.

SYMBOL	DESCRIPTION
K	the number of topics
D	the total number entities
D_1, D_2	the number of entities in source 1 and source 2
$l_{d,d'}$	the value in the matching relation matrix, denotes whether d is matched with d'
$w_{d,j}$	the jth attribute (word) in entity d
$z_{d,j}$	the topic assigned to attribute $w_{d,j}$
$c_{d,j}$	the latent variable assigned to $z_{d,j}$, the value of $c_{d,j}$ can be d or the index of matched entities with d
θ_d	multinomial distribution over topics specific to entity d
$\varphi_{1,z}$, $\varphi_{2,z}$	multinomial distribution over terminologies specific to topic z in source 1 and 2
λ_d	multinomial distribution over latent variables c specific to entity d
α, β	Dirichlet priors to multinomial distributions θ and λ
γ	global regression parameter
ρ	a function provides binary probabilities used to generate $l_{d,d'}$
e_1, e_2	two constant values used to determine β (weights of the prior for cross-sampling)

3.1 Model Overview

Framework. The basic assumption of the proposed model is that, for entities from different sources, *their matching relations and hidden topics are influenced by each other*. Matching entities are similar in hidden space of topics, though the topics have different representations (e.g., word distributions) in different sources, and vice versa, entities that are similar in hidden space of topics tend to be matched. Thus the basic idea here is *to leverage the known matching relations to help the extraction of hidden topics, and use the extracted topics to infer the unknown relations*.

Figure 2 shows the plate representation of the proposed semi-supervised model. For simplicity, we omit the modeling part for the words in source 2 as it is the same as source 1. Table 1 summarizes the notations used in the CST model.

In order to avoid pairwise relation modeling, before we use CST to model the generation of given entities and the generation of matching relations, we first process a candidate filtering. For the entities that have no chance to be matched with each other, CST will not model the relation generation for them. For example, given

Input: a dual source corpus C, a matching relation matrix L, and hyper-parameters α and β
foreach *entity* d **do**
  Generate $\theta_d \sim \text{Dir}(\alpha)$;
end
% cross-sampling-based entity generation
foreach d *in each source* t **do**
  Set β according to $\mathbf{L}_d$;
  Generate $\lambda_d \sim \text{Dir}(\beta)$;
  for $n = 1$ *to* N_d **do**
    Generate $c_{d,n} \sim \text{Mult}(\lambda_d)$, $c_{d,n}$ can be d or the index of matched entities with d;
    Draw a topic $z_{d,n} \sim \text{Mult}(\theta_{c_{d,n}})$ from the topic distribution of the entity c;
    Draw a word $w_{d,n} \sim \text{Mult}(\varphi_{t,z_{d,n}})$ from $z_{d,n}$-specific word distribution;
  end
end
% matching relation generation
foreach (d, d') *with possible links* **do**
  Generate $l_{d,d'} \sim \rho(\cdot|\boldsymbol{z}_d, \boldsymbol{z}_{d'}, \gamma)$;
end

Algorithm 1: Generative process for the CST model

a Wiki article describing a product (e.g., iPhone), we only consider patents belonging to the company which creates this product (e.g., Apple) in relation generation part. More general method to filter candidates is left as future work.

Cross-Sampling. We then introduce an important concept in the CST model: cross-sampling, which allows CST to leverage known relations and extract cross-source topics. The idea of cross-sampling is: when generating topics for an entity d, the sampling process is not only based on the topic distribution of d, but also the topic distributions of all the matching entities of d. The intuition behind the idea is that the matched entities are similar in hidden space of topics. For example, a user would like to edit a Chinese Wikipedia article about "Barack Obama." Before he starts, he may take a look at what topics the corresponding English Wikipedia article contains, and finds out that the article contains Obama's early career as a Chicago community organizer. Thus he will edit the Chinese Wikipedia article to present Obama's experience as a community organizer but in different words. This process of cross-sampling allows us to bridge the topics in entities from different sources and model the cross-source topics.

By cross-sampling, the CST model utilizes the known matching relations and makes the matching entities to have similar topic distributions. Similar ideas are proposed in some other models [8, 10, 17]. However, these unsupervised methods can hardly infer unknown relations in a unified model. As we will introduce later, CST employs a semi-supervised learning algorithm to infer unknown relations. Another kind of linked topic models [7, 20, 21] are able to infer missing links between entities. However, they do not consider the direct effect of known links on hidden topics, and CST employs cross-sampling to model a more explicit and high-order dependency between matching entities. The more sufficient utilization of known relations makes the CST model more suitable for heterogeneous source corpses than traditional topic models (experiments show that the CST model outperforms RTM, a traditional linked topic model, by 40.9% on average).

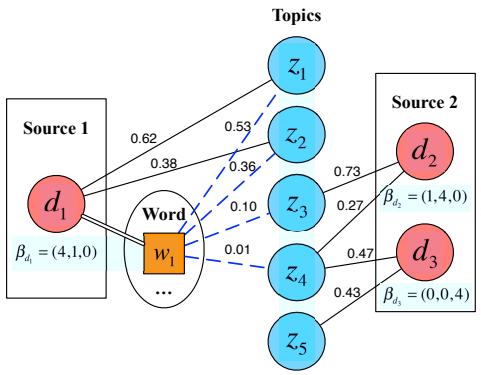


Figure 3: An intermediate step of cross-sampling. There is a matching relation between d_1 from source 1 and d_2 from source 2. Latent topics for a word w_1 from d_1 is sampled based on both d_1 and d_2.

3.2 Generative Process

Formally, the generative process is described in Algorithm 1. It consists of two parts: (1) cross-sampling-based entity generation and (2) matching relation generation.

Cross-Sampling-Based Entity Generation. Here, we introduce the entity generation in detail. First, for each entity d in source 1, we sample its topic distribution θ_d: $\theta_d \sim \text{Dir}(\alpha)$. Next, for each word w in d, we choose a topic z: $z \sim \text{Mult}(\theta_c)$, where c could be d itself or one of d's matching entities. We sample c according to $c \sim \text{Mult}(\lambda_d)$, where λ_d indicates how likely an entity matched with d (including d itself) will be sampled. λ_d is sampled according to $\lambda_d \sim \text{Dir}(\beta_d)$, β_d is a $|D|$-dimensional vector, where $|D|$ is the total number of entities, and we define β_d as follows: we set $\beta_{d,d} = e_1$, where e_1 is a constant value denotes the weight of the prior to sample d's topics from its own topic distribution θ_d; for an entity d' matched with d, we set $\theta_{d,d'} = e_2$, where e_2 is another constant value represents the weight of the prior to sample topics from one of d's matching entities; for other entities we set the corresponding values in β to 0.

Figure 3 gives an example, in which we have three entities d_1, d_2, and d_3; d_1 is from source 1; d_2 and d_3 are from source 2; the only matching relation exists between d_1 and d_2. Thus we set $\beta_{d_1} = (e_1, e_2, 0)$. From Figure 3, we can see that d_1 is only assigned with topics z_1 and z_2 in last step. However, in this step, as there is a matching relation between d_1 and d_2, the word w_1 from d_1 can still be assigned with topics from d_2 (z_3 and z_4), which bridges the latent topic space between linked entities.

With above definition, there is no chance to sample an entity d's topics from entities not matching with d. If d has no matching relations, each z is sampled according to its own entity's topic distribution θ_d. Thus the generation of d is the same with LDA [4].

Finally the word w is sampled according to the word distribution of topic z in source 1: $w \sim \text{Mult}(\varphi_{1,z})$. As different terminologies are used to represent the same topic in different sources, we separate the word distribution of a topic z into $\varphi_{1,z}$ and $\varphi_{2,z}$. We use source 1 as an example above and the documents in source 2 are generated in the same way.

Matching Relation Generation. In this step, each matching relation $l_{d,d'}$ is modeled as a binary variable. As entities with similar topic distributions tend to be matched with a higher probability, it is natural to model the probability of a matching relation as a func-

tion ρ of topic distributions. There are many possibilities for the function ρ. In this paper, we consider the following form

$$\rho(l_{d,d'} = 1|\boldsymbol{z}_d, \boldsymbol{z}_{d'}, \gamma) \propto \exp[\boldsymbol{\gamma}^T(\tilde{\boldsymbol{z}}_d \circ \tilde{\boldsymbol{z}}_{d'})] \tag{1}$$

where the $\circ$ notation denotes the Hadamard product $((\tilde{\boldsymbol{z}}_d \circ \tilde{\boldsymbol{z}}_{d'})_k = \tilde{z}_{d,k} \times \tilde{z}_{d',k})$, $\tilde{\boldsymbol{z}}_d$ is a K-dimension vector indicating the appearance of each topic in d, $\tilde{z}_{d,k} = \sum_{j=1}^{N_d} \mathbf{1}(z_{d,j} = k)$. The function ρ is parameterized by coefficients $\boldsymbol{\gamma}$. We define the function as an exponential one thus when $\boldsymbol{z}_d$ and $\boldsymbol{z}_{d'}$ are close, with large weighted Hadamard product, the probability increases exponentially.

A similar regression method is used in Relational Topic Model (RTM) [7]. The difference between RTM and CST is, RTM can hardly deal with the entities from multiple sources while CST bridges multiple sourced entities by learning how likely they will be influenced by each other (λ). Also, by cross-sampling, CST models a high-order dependency between matching entities and utilize the known relations more sufficiently.

As a conclusion, cross-sampling-based entity generation allows CST to leverage the known relations to help extract hidden cross-source topics. The matching relation generation uses extracted topics to infer the relations between entities in a latent space.

3.3 Model Learning

According to the model description above, the likelihood of the observed data in the CST model is given as

$$\begin{aligned} P(\boldsymbol{w}, L|\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\varphi}) = &\prod_{d_1,d_2} (\sum_{z_{d_1}, z_{d_2}} P(l_{d_1,d_2}|z_{d_1}, z_{d_2}, \boldsymbol{\gamma})) \\ &\times \int_{\boldsymbol{\theta}} \{\prod_{d}^{D} \int_{\lambda_d} [P(\theta_d|\boldsymbol{\alpha}) P(\lambda_d|\boldsymbol{\beta}) \prod_{j=1}^{N_d} \sum_{c_{d,j}} (P(c_{d,j}|\lambda_d) \\ &\times \sum_{z_{d,j}} (P(z_{d,j}|\boldsymbol{\theta}, c_{d,j}) \times P(w_{d,j}|z_{d,j}, \boldsymbol{\varphi})))] d\lambda_d \} d\boldsymbol{\theta} \end{aligned} \tag{2}$$

where $\boldsymbol{w}$ is a set of observed words in given corpus, L is the matching relation matrix, d_1 and d_2 are two entities with a labeled $l_{d,d'}$ ($l_{d_1,d_2} \neq ?$), and N_d is the number of words in entity d.

We employ MAP estimation to learn the parameters of the CST model. However, the exact posterior inference is intractable and we appeal to approximate inference methods. In this work, we employ the mean-field variational inference [28, 12]. Generally, we define four variational parameters and aim to maximize the evidence lower bound (ELBO) [30]. Specifically, We define $\boldsymbol{\vartheta}$ and $\boldsymbol{\epsilon}$ as variational multinomial parameters. We also define $\boldsymbol{\tau}$ and $\boldsymbol{\eta}$ as variational Dirichlet parameters. The approximate posterior is then defined as

$$\begin{aligned} &Q(\boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{c}|\boldsymbol{\vartheta}, \boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\epsilon}) = \\ &\prod_{d=1}^{D} q_\theta(\theta_d|\tau_d) q_\lambda(\lambda_d|\eta_d) \prod_{n=1}^{N_d} q_z(z_{d,n}|\vartheta_{d,n}) q_c(c_{d,n}|\epsilon_d) \end{aligned} \tag{3}$$

We aim to minimize the Kullback-Leibler (KL) divergence between the variational distribution and the true posterior, which is equivalent to maximizing the evidence lower bound (ELBO) [30]. The complete equation of ELBO is shown below.

$$\begin{aligned} L(\boldsymbol{\vartheta}, \boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\epsilon}) = &\sum_{d_1,d_2} \mathbb{E}_q[\ln P(l_{d_1,d_2}|\boldsymbol{z}_{d_1}, \boldsymbol{z}_{d_2}, \gamma)] + \sum_{d=1}^{D} \mathbb{E}_q[\ln P(\theta_d|\alpha)] \\ &+ \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\ln P(c_{d,n}|\lambda_d)] + \sum_{d=1}^{D} \mathbb{E}_q[\ln P(\lambda_d|\beta)] \\ &+ \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\ln P(z_{d,n}|\boldsymbol{\theta}, c_{d,n})] + \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\ln P(w_{d,n}|z_{d,n}, \varphi)] \\ &- \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\ln q_z(z_{d,n})] - \sum_{d=1}^{D} \sum_{n=1}^{N_d} \mathbb{E}_q[\ln q_c(c_{d,n})] \\ &- \sum_{d=1}^{D} \sum_{i=1}^{K} \mathbb{E}_q[\ln q_\theta(\theta_{d,i})] - \sum_{d=1}^{D} \sum_{d' \in R(d)} \mathbb{E}_q[\ln q_\lambda(\lambda_{d,d'})] \end{aligned} \tag{4}$$

where d_1 and d_2 stratify $l_{d_1,d_2} \neq ?$. We then need to compute each item in Eq. 4. We focus on the first item as others, which are expected values of the log of a single probability component under the Dirichlet or the multinomial, can be expanded similar with LDA model. The first term is:

$$\begin{aligned} &\mathbb{E}_q[\ln P(l_{d_1,d_2}|\boldsymbol{z}_{d_1}, \boldsymbol{z}_{d_2}, \gamma)] = \mathbb{E}_q[\gamma(\boldsymbol{z}_{d_1} \circ \boldsymbol{z}_{d_2})] \\ &= \gamma(\frac{\sum_{n=1}^{N_{d_1}} \phi_{d_1,n}}{N_{d_1}} \circ \frac{\sum_{n=1}^{N_{d_2}} \phi_{d_2,n}}{N_{d_2}}) \end{aligned} \tag{5}$$

We then take the derivatives with respect to each variational parameter. We use η as an example. We first collect all of the terms associated with $\boldsymbol{\eta}$ and get:

$$\begin{aligned} L_{[\eta]} = &\sum_{d=1}^{D} (\sum_{c \in R(d)} (N_d \times \epsilon_{d,c} + \beta_{d,c} - \eta_{d,c})(\Psi(\eta_{d,c}) - \Psi(\sum_{i \in R(d)} \eta_{d,i})) \\ &- (\log \Gamma(\sum_{i \in R(d)} \eta_{d,i}) - \sum_{c \in R(d)} \log \Gamma(\eta_{d,c})) \end{aligned}$$

We then take the derivative with respect to $\boldsymbol{\eta}$

$$\frac{\partial L_{[\eta]}}{\partial \eta_{d,c}} = (N_d \times \epsilon_{d,c} + \beta_{d,c} - \eta_{d,c})(\Psi'(\eta_{d,c}) - \Psi'(\sum_{i \in R(d)} \eta_{d,i}))$$

The derivations of other variational parameters could be obtained similarly. We then set the derivations to zero, and find:

$$\eta_{d,c} = \beta_{d,c} + N_d \times \epsilon_{d,c} \tag{6}$$

$$\tau_{d,k} = \alpha_k + \sum_{n=1}^{N_d} \vartheta_{d,n,k} \tag{7}$$

$$\epsilon_{d,n,c} \propto \exp\{\Psi(\eta_{d,c}) - \Psi(\sum_{i \in R(d)} \eta_{d,i})\} \tag{8}$$

$$\begin{aligned} \vartheta_{d,n,k} \propto &\sum_{d' \in \{R(d),d\}} (\exp\{\sum_{d'' \neq d'} \frac{\gamma_k \sum_{i=1}^{N_{d''}} \vartheta_{d'',i,k}}{N_{d'} N_{d''}} \\ &+ \Psi(\tau_{d',k}) - \Psi(\sum_{j=1}^{K} \tau_{d',j})\} \epsilon_{d,n,d'} \times \varphi_{t,k,v}) \end{aligned} \tag{9}$$

where t is the source of entity d, v is the n-th word of d, and $R(d)$ is a set of entities matched with d. Intuitively, Eq. 9 utilizes the

known relations to update ϑ. The first summation in this equation is related with cross-sampling and the second one is based on the regression part of CST. These updates above are performed iteratively until convergence, since they depend on each other.

We then fit the model by maximizing the resulting ELBO with respect to the model parameters $\boldsymbol{\varphi}$ and $\boldsymbol{\gamma}$. In source t, given a topic k and a term v, the update for $\varphi_{t,k,v}$ is:

$$\varphi_{t,k,v} \propto \sum_{d=1}^{D_t} \sum_{n=1}^{N_d} \vartheta_{d,n,k} \mathbf{1}(w_{d,n}^t = v) \tag{10}$$

The derivate with respect to $\boldsymbol{\gamma}$ takes a convenient form. To solve this problem, we add a 2-norm regularizer, which penalizes the objective function with the term $\zeta||\lambda||_2$, where ζ is a free parameter. We then have:

$$\gamma_k = \frac{\sum_{d,d'} 1}{2\sum_{d,d'} l_{d,d'}[(\Upsilon_d - \Upsilon_{d'}) \circ (\Upsilon_d - \Upsilon_{d'})]_k} \tag{11}$$

where d and d' are two entities with a labeled $l_{d,d'}$ ($l_{d,d'} \neq ?$), and $\Upsilon_{d,k} = \frac{\sum_{n=1}^{N_d} \vartheta_{d,n,k}}{N_d}$. Both the above update and Eq. 9 utilize known relations.

With all update equations above, we employ the variational expectation-maximization algorithm to learn the model, which yields the following iterations: (See Algorithm 2 for details.)

E-step: optimize the ELBO with respect to the variational parameters $\{\boldsymbol{\vartheta}, \boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\epsilon}\}$. Update these variational parameters according to Eqs. 6-9.

M-step: maximize the resulting ELBO with respect to the model parameters $\{\boldsymbol{\varphi}, \boldsymbol{\gamma}\}$. Update the model parameters according to Eqs. 10-11.

Inferring Matching Relations. We finally detect the matching entities from different sources. Given a dual source corpus and a matching relation matrix with missing values, we use the learning algorithm from Section 3.3 to estimate the model's parameters by optimizing the ELBO for the observed data: words from the corpus and known relations in the matching relation matrix. After that, given two entities d and d' with an unknown relation ($l_{d,d'} = ?$), we use the fitted model's variational parameters to approximate the predictive probability:

$$P(l_{d,d'}|\boldsymbol{w}_d, \boldsymbol{w}_{d'}) \approx \mathbb{E}_q[p(l_{d,d'}|\boldsymbol{z}_d, \boldsymbol{z}_{d'})] \tag{12}$$

The right hand of Eq. (12) is an expectation of ρ (defined in Eq. 1) with respect to the approximation posterior (Eq. 3). Intuitively, the approximated predictive probability indicates that CST considers the content information and infers the matching relations between entities in hidden space of topics. Also, CST can be plugged into other detection frameworks (e.g., random walk [15] or factor graphs [14]) easily, to further leverage structural information. Details and two examples will be described in the next section.

4. EXPERIMENTS

We evaluate our proposed model with two experiments. All datasets and codes used in this work are publicly available[2].

[2]http://arnetminer.org/document-match/

Input: a dual source corpus C, a matching relation matrix L, and hyper-parameters α and β
Initialize $\{\boldsymbol{\vartheta}, \boldsymbol{\tau}, \boldsymbol{\eta}, \boldsymbol{\epsilon}, \boldsymbol{\varphi}, \boldsymbol{\gamma}\}$ randomly;
repeat
  % E-Step: optimize the ELBO;
  foreach d *in each source* t **do**
    for $c = 0$ *to* 1 **do**
      Update $\eta_{d,c}$ according to Eq. 6;
    end
    for $k = 1$ *to* K **do**
      Update $\tau_{d,k}$ according to Eq. 7;
    end
    for $n = 1$ *to* N_d **do**
      for $c = 0$ *to* 1 **do**
        Update $\epsilon_{d,n,c}$ according to Eq. 8;
      end
      for $k = 1$ *to* K **do**
        Update $\vartheta_{d,n,k}$ according to Eq. 9;
      end
    end
  end
  % M-Step: maximize the resulting ELBO;
  foreach *topic* k *in each source* t **do**
    foreach *term* v **do**
      Update φ according to Eq. 10;
    end
    Update γ_k according to Eq. 11;
  end
until *Convergence*;

Algorithm 2: Variational EM for model learning.

4.1 Tasks and Data Sets

We validate the proposed model in two real scenarios: product-patent matching and cross-lingual matching. We describe the details of each task below.

Product-patent matching. In this task, given a Wiki article describing a specific product, we aim to find relevant patents, e.g., a Wiki article and a patent should be relevant if they are both talking about the topic of Siri. We collect 13,085 Wiki articles and 15,000 patents from Wikipedia and USPTO respectively. For some Wiki article that describes a product, we use it as a query to find patents related with the same product. One Wiki article may be matched with more than one patent, e.g., a Wiki article describing iPhone corresponds to patents that claim on touch screen, camera, soft keyboard, etc.. We sample 233 Wiki articles as queries and find 1,060 matching relations in total. We randomly choose 30% of the matching relations as known. The remaining relations are regarded as unknown and need to be inferred.

The ground truth data, which consists of 1,060 matching relations, is labeled by four human annotators. For each of 233 Wiki articles as queries, each annotator reads all patents belonging to the same company with the corresponding product in the query. Some online systems and materials are referred when filtering the candidates and labeling the data (e.g., PatentMiner [25][3], news related with companies' lawsuit, official documents of the products, etc.). To see more details of how we label the data, please refer to our public web page[2]. We say a Wiki article is matched with a patent when four annotators all agree. Based on this work, we have deployed a product-patent matching function to PatentMiner. We are

[3]A public patent search and analysis system: http://pminer.org

Table 2: Performance of product-patent matching task.

Method	P@3	P@20	MAP	R@3	R@20	MRR
CS + LDA	0.111	0.083	0.109	0.011	0.046	0.053
RW + LDA	0.111	0.117	0.123	0.033	0.233	0.429
RTM	0.501	0.233	0.416	0.057	0.141	0.171
RW + CST	**0.667**	0.167	0.341	**0.200**	0.333	0.668
CST	**0.667**	**0.250**	**0.445**	0.171	**0.457**	**0.683**

Table 3: Performance of cross-lingual matching task.

Method	Precision	Recall	F_1-Measure	F_2-Measure
Title Only	**1.000**	0.410	0.581	0.465
SVM-S	0.957	0.563	0.709	0.613
LFG	0.661	0.820	0.732	0.782
LFG + LDA	0.652	0.805	0.721	0.769
LFG + CST	0.682	**0.849**	**0.757**	**0.809**

collecting user feedbacks to create a bigger evaluation data set for future work.

Cross-lingual matching. In this task, given an English Wiki article, we aim to find a Chinese article, which reports the same content, from a Chinese Wiki knowledge base. We use the same data set with [29]. The data set is collected as follows: we first randomly select an English article A with a cross-lingual link to a Chinese article B from Wikipedia. We then use the B's title to find another Chinese article C with the same title in Baidu Baike[4]. As A is cross-lingually linked with B in Wikipedia, and B has the same main idea with C (normally a Wiki article uses its main idea as the title). It is reasonable to say there is a cross-lingual matching relation between A and C.

The data set consists of totally 2,000 English articles from Wikipedia, and 2,000 Chinese articles from Baidu Baike. Each English article corresponds to one Chinese article. We conduct 3-fold cross validation on the evaluation data set.

4.2 Evaluation

Evaluation metrics. In the first experiment, for each Wiki article, we rank all patents according to the probability predicted by the proposed model and alternative methods. We evaluate all the methods in terms of P@3 (Precision for the top 3 ranking results), P@20, MAP (Mean Average Precision), R@3 (Recall for the top 3 results), R@20, and MRR (Mean Reciprocal Rank).

In the second experiment, to keep consistence with [29], we consider cross-lingual matching as a two-class classification problem: given an English Wiki article and a Chinese Wiki article, we label this pair of two entities as "matched" or "not matched". We compare all baselines in terms of Precision (Prec.), Recall (Rec.), F_1-Measure (F_1), and F_2-Measure (F_2).

Comparison methods. For the first experiment, we compare the following methods for product-patent matching:

• ***Content Similarity based on LDA (CS + LDA):*** It calculates the similarity between a Wiki article and a patent based on their topic distributions calculated by LDA. Specifically, we use p_{d_1} and p_{d_2} to represent the topic distribution of a Wiki article and a patent respectively. The similarity score is defined based on the Cosine similarity between p_{d_1} and p_{d_2}

$$Sim(d_1, d_2) = \frac{p_{d_1} \cdot p_{d_2}}{||p_{d_1}|| \times ||p_{d_2}||} \tag{13}$$

[4]A Chinese Wiki knoledge base: http://baike.baidu.com/

• ***Random Walk based on LDA (RW + LDA):*** It ranks candidates by combining the extracted topics into a random walk with restart algorithm [27]. Specifically, it creates a graph containing Wiki articles and patents as nodes. And it links a Wiki article u to a patent v with a weight

$$W_{u,v} = \begin{cases} \frac{Sim(u,v)}{\sum_w Sim(u,w)} & \text{if } Sim(u,v) \geq \mu \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where μ is a threshold value defined manually, and $Sim(u, v)$ is the Cosine similarity between u and v. Thus there is a bigger chance for a Wiki article node to reach a more similar patent node. It employs LDA to calculate the topic distributions. Besides the textural contents of entities, this framework also considers the structural information. We create a link from one patent node to another if the former one cites the latter one. We also create a link from one Wiki article nodes to another if they have a hyperlink in Wikipedia. The weights of these links are defined as a constant value (in practice, we define all of them as 1). Finally, the transition probability from u to v can be defined as

$$Q_{u,v} = (1 - a)\frac{W_{u,v}}{\sum_x W_{u,w}} + a\mathbf{1}(v = s) \tag{15}$$

where s is the start node, a is the restart probability.

• ***Relational Topic Model (RTM):*** It employs the RTM, which is generally used to model the links between entities, proposed by Blei et al. [7]. In our problem, this method regards there is a link between two matching entities. We use Blei's implementation of RTM[5].

• ***Random Walk based on CST (RW + CST):*** The difference between this method and RW + LDA is, instead of using $Sim(u, v)$ to define the weight of links from a Wiki node to a patent node, it uses $P(l_{u,v})$ (see Section 3.3 for details) calculated by CST.

• ***CST:*** It is our proposed model. We first use the training set to learn the model. Then we use the fitted model to detect unknown relations. We set $K = 50$, $\alpha = 50/K$, $e_1 = 4$, and $e_2 = 1$ in both this method and RW + CST.

All methods use entities in the training set to fit the model. Methods related to RTM or CST utilize known matching relations as guidance, while LDA is unable to leverage this information. Random walk based methods further consider structural information (citations in the patent database and hyperlinks in Wikipedia).

For the second experiment, we compare the following methods for cross-lingual matching:

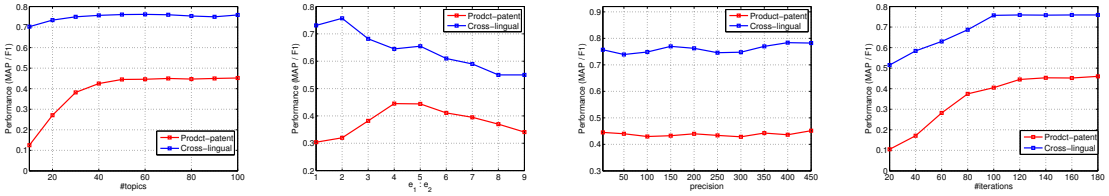
• ***Title Only:*** This method first translates the title of Chinese articles into English by Google Translation API [6], then matches the translated titles with English articles. Two articles are considered as equivalent ones if they have strictly the same English titles.

• ***SVM-S:*** It is a classifier proposed by Sorg et al. [24] to find cross-lingual links between English Wikipedia and German Wikipedia. The authors define several graph-based and text-based features. Here we train a SVM with their features on evaluation data set. For SVM, we choose LIBSVM [6].

• ***LFG:*** It is the method proposed by Wang et al. [29], which is based on a factor graph model and mainly considers the structural information to solve the problem of cross-lingual matching.

[5]http://www.cs.princeton.edu/~blei/topicmodeling.html

[6]https://developers.google.com/translate/?hl=zhcn



(a) Number of topics K (b) Ratio (c) Precision (d) Convergence analysis

Figure 4: Parameter analysis. (a) Performance of the CST model by varying the number of topics K; (b) Performance of the CST model by varying the ratio of e_1 to e_2; (c) Performance of the CST model is stable when varying the precision of β; (d) Convergence analysis of the CST model. Y-axises in all figures denote the MAP value of the CST model in product-patent matching experiment and the F1 score of the CST model in cross-lingual matching experiment.

• ***LFG + LDA:*** It adds a feature, which captures the content similarity between articles, to the feature function of LFG. It uses $Sim(u, v)$ (see Eq. 14) as the feature value.

• ***LFG + CST:*** LFG mainly considers structural information. We enhance it by bringing in content information (hidden topics extracted by CST). The difference between this method and LFG + LDA is that, instead of using $Sim(u, v)$ to define the newly added feature, it uses $P(l_{u,v})$ calculated by CST. We compare this method with LFG to see if content information can help in this problem. We compare it with Title Only and SVM-S to show the power of utilizing corss-lingual topics extracted by CST. We also compare it with LFG + LDA to show the effectiveness of the CST model compared with a traditional topic model. Here we keep values of K, α, and e_2 the same with the first task, and set $e_1 = 2$. We will give the intuitive explanation why we change e_1 latter.

4.3 Quantitative Results

Product-patent matching. Table 2 lists the performance of product-patent matching problem using different methods. We first compare CST with two unsupervised methods, CS + LDA and RW + LDA. With the help of known relations as guidance, we can see CST clearly outperforms these two methods (+72.4%-75.5% in terms of MAP). We then compare CST with RTM, which also utilizes the known relations as guidance. With the help of the cross-sampling, CST can better extract cross-source topics. Thus it can better detect the matching relations (+74.9% in terms of MRR). To our surprise, when employing the CST model, combining content and structural information hurts the performance (RW + CST drops 23.4% in terms of MAP). By a careful investigation, we find that a Wiki article normally has lots of hyperlinks to other articles (56.4 out-links in average). Much noise is contained in these links and hurts the performance. However, the structural information does help for top results (+14.5% in terms of R@3).

Cross-lingual matching. Table 3 shows the performance of cross-lingual matching problem. Title Only and SVM-S employ the translated terminologies and perform well in terms of Prec. However, without capturing the hidden topics of entities, the translation can not be performed precisely. Thus these methods miss a number of matching relations between entities, which hurts the Recall.

LFG focuses on utilizing structural information. We enhance this method by bringing in hidden topics extracted by LDA and CST respectively. From the table, we see that LFG + CST improves the performance. It outperforms all baselines in terms of Recall, F_1, and F_2 (e.g., averagely +15.2% in terms of F_2). In fact, cross-lingual topics can hardly be extracted due to the low co-occurrence of English and Chinese terminologies. Without a precise cross-lingual topic extraction, LFG + LDA performs worse than LFG, which indicates the incorrect topics will hurt the performance. By studying some cross-lingual topics found by the CST model, we find that the top Chinese and English terminologies in the same topic are very relevant. Some Chinese terminologies are translated results of English ones.

Topics analysis. How many topics are enough for the product-patent matching problem and cross-lingual matching problem? We perform an analysis by varying the number of topics in the CST model. Figure 4(a) shows its performance with number of topics K varied. We can see that, the performance improves by increasing K when K is small (< 50). After that, the trend becomes stable.

Ratio analysis. We study how the ratio of e_1 to e_2 influence the performance. We fix e_2 as 1 and vary e_1. Figure 4(b) shows the trend of the performance following the changes of the ratio in both two problems. In the product-patent matching problem, the value of MAP reaches largest when $e_1 : e_2 = 4$. And in the cross-lingual problem, F1 reaches the maximum value when $e_1 : e_2 = 2$, corresponding to a larger prior probability of cross-sampling.

Intuitively, compared with cross-lingual matched articles, patents and Wiki articles with matching relations are more dissimilar in hidden space of topics: patents focus on specific technologies, while Wiki articles describe general descriptions of products (e.g., histories, sales, etc.). And cross-lingual matched articles report the same objects. Thus the prior of cross-sampling in cross-lingual matching problem should be larger (smaller e_1, larger e_2). It indicates that the hyper-parameters of CST can be determined intuitively: if the matching entities in a specific problem assumed to be more similar in topics, we can give a smaller value to $e_1 : e_2$, otherwise we should set $e_1 : e_2$ a larger value.

Precision analysis. We further investigate how the precision [19] of β, which indicates the confidence in the prior, influence the performance. We vary the precision from 1 to 450. As Figure 4(c) shows, the CST model's matching performance is not sensitive to the precision of β.

Convergence analysis. We finally investigate the convergence of the CST model. Figure 4(d) shows the convergence analysis of the CST model on product-patent matching problem and cross-lingual matching problem. We see the CST model converges within 100 iterations ion both two tasks.

4.4 Qualitative Results

In this section, we demonstrate some examples generated from our experiments to show the effectiveness of the CST model.

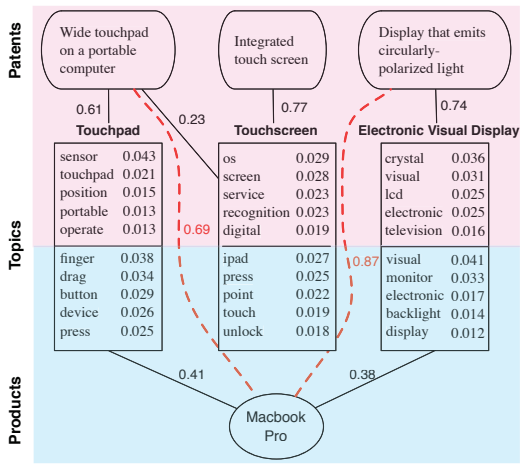


Figure 5: Examples of the correlations between topics, patents, and Wiki articles in the CST model. θ, the probability of a topic give an entity, is represented on each black-solid edge. And the weight on each red-dotted edge denotes the likelihood of a matching relation. The titles of topics are hand-labeled. And for each topic, we separate the terminologies used in patents (the upper part of each topic box) and the terminologies used in Wiki articles (the lower part of each topic box). We remove some edges whose probabilities are negligible.

Table 4: Examples of topics highly relevant to both Apple and Samsung found by the CST model. Top terminologies from each source are showed. The titles of topics are hand-labeled.

Title	Top Patent Terms	Top Wiki Terms
Gravity Sensing	rotational, gravity, interface, sharing, frame, layer	gravity, iPhone, layer, video, version, menu
Touchscreen	recognition, point, digital, touch, sensitivity, image	screen, touch, iPad, os, unlock, press
Application Icons	interface, range, drives, icon, industrial, pixel	icon, player, software, touch, screen, application

Product-patent matching. Figure 5 shows a part of the matching results of "Macbook Pro" Wiki article. We select 3 topics extracted by the CST model and display them with top words in both two sources. We also represent the probability of a specific topic z given an entity d ($\theta_{z,d}$), and the matching probability of two entities in the form of edges. As we can see from the figure, a patent mostly focus on one topic, a specific technology. And a Wiki article generally describe a number of features of a product. Thus Wiki articles have more diverse topic distributions.

When predicting a matching relation for two entities, the regression part of the CST mode is able to distinguish relevant topics from others. As the figure shows, the CST mode successfully detects the Macbook Pro is matched with "Wide touchpad on a portable computer" and "Display that emits circularly-polarized light" respectively. Each of the two patents is associated with a topic relevant to Macbook Pro.

Apple vs. Samsung. The CST model will be helpful to find the patents that a company uses to protect her products, by detecting the matching relations between products and patents. CST is also able to infer the inner connections between companies. Given a set of companies, we train the CST model by these companies' patents and Wiki articles describing the companies' products. And for a company g, we define its topic distribution as $P(z|g) = \frac{\sum_{d=1}^{D_g} \theta_{d,z}}{\sum_{d=1}^{D_g} 1}$, where D_g is the set of entities relevant to g. Here we use Apple and Samsung as an example. Table 4 lists the top three topics related with both Apple and Samsung. We also represent each topic's top words from both Wiki articles and patents. We see that terminologies related with technologies are more likely to appear in patents (e.g., recognition, range, etc.). And most terms closer to our lives and applications are from Wikipedia (e.g., video, iPad, etc.).

"Gravity Sensing" and "Touchscreen" are both highly related with the products of Apple and Samsung (e.g., smart phones, iPad, etc.), which indicates through the label information between patents and products, the CST model can identify the topics bridging products and related technologies. Moreover, "Application Icons" is also discovered by CST. As we know, one of the Apple patents been violated by Samsung[7], is the design patent 305: Rounded square icons on interface, which is related to this topic. It indicates that the results of CST may be helpful to infer the competitive relationships between companies.

5. RELATED WORK

Cross-source matching. We first review some related work on cross-source matching problem. Wang et al. [29] study the cross-lingual knowledge linking problem. They aim to link Chinese and English Wiki articles which report on the same content. However, the model they proposed, called LFG, only considers the structural information. In this paper, we utilize our proposed model to bring in content information to LFG. We conduct a similar experiment with Wang et al. and the result shows that the performance is significantly improved. Mimno et al. [18] has studied a similar problem. They propose a polylingual topic model to discover topics aligned across multiple languages. Tang et al. [26] propose a method called Cross-domain Topic Learning. Their goal, which is to recommend cross-domain collaborations, is different from ours. More importantly, their method separates topic extraction and link prediction into two models. Our model integrates *topic modeling* and *entity matching* into a unified model.

Besides, Barnard et al. [2] propose an approach for modeling segmented images with associated text simultaneously. However, their approach do not integrate entity matching and topic modeling into a uniform framework.

Topic modeling. It is natural to apply topic modeling (e.g., LDA [4] and PLSA [11]) on a collection of documents, and use the derived topic distribution to represent each document. The basic mechanism behind these models is to exploit co-occurrence patterns of words in documents to find K semantically meaningful topics and best describe the given corpus. However, both PLSA and LDA treat documents in a given corpus independently.

To deal with the pairwise information of documents, Cohn and Hoffman [10] build an extension to the PLSA model, which is called PHITS. A similar model called mixed membership model is developed by Erosheva et al. [9]. Mei et al. [17] add a regularization constraint on a prior knowledge that some pairs of documents should be similar, to the traditional topic models. Dietz et al. [8, 16] also propose similar methods. These approaches regard links as input data, whereas in our work, the proposed model is able to infer the unknown relations between documents from different sources.

[7]http://www.businessinsider.com/apple-versus-samsung-2012-8

To integrate supervised information, Blei et al. propose a method called Relational Topic Model (RTM) [7], which models the links of each pair of documents as a binary random variable that is conditioned on their contents. Nallapati et al. [20] propose a model combines the ideas of LDA and Mixed Membership Block Stochastic Models [1] and allows modeling arbitrary link structure. They also propose another model [21], which assumes the link structure is a bipartite graph and combines the LDA and PLSA into a single graphical model. Blei et al. [5] introduce supervised LDA and use it to predict ratings for movies. All these models introduced above depend on the co-occurrence of terms, whereas multiple source documents (entities) share few terms. Thus they can hardly deal with corpus from different sources. By cross-sampling, our proposed model is able to extract cross-source topics and infer matching documents (entities) from different sources.

6. CONCLUSION

In this paper, we propose an approach to solve the problem of entity matching across heterogeneous sources. The model we proposed is named as the Cross-Source Topic model, which integrates the topic extraction and entity matching into a unified framework. A semi-supervised learning algorithm is proposed to learn the model. We validate the model on two real scenarios. The experimental results demonstrate that the proposed model can extensively improve the performance compared with baseline methods (+19.8% and +7.1% in two scenarios respectively). The proposed model mainly considers the text content information of entities. Meanwhile, the model is easy to be plugged into other frameworks which can leverage both the content and structural information together. We give two examples to show how the proposed model can be plugged into a random walk framework and a factor graph respectively.

Acknowledgements. The work is supported by the National High-tech R&D Program (No. 2014AA015103), National Basic Research Program of China (No. 2014CB340506, No. 2012CB316006), Natural Science Foundation of China (No. 61222212), National Social Science Foundation of China (No. 13&ZD190), NSF CAREER Award (No. 1453800), NSFC-ANR (No. 61261130588), and a research fund supported by Huawei Inc.

7. REFERENCES

[1] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. *JMLR*, 9:1981–2014, 2008.

[2] K. Barnard, P. Duygulu, D. Forsyth, N. De Freitas, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 3:1107–1135, 2003.

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

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

[5] D. M. Blei and J. D. McAuliffe. Supervised topic models. *arXiv preprint arXiv:1003.0783*, 2010.

[6] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2:27:1–27:27, 2011.

[7] J. Chang and D. Blei. Relational topic models for document networks. In *AIS'09*, pages 81–88, 2009.

[8] L. Dietz, S. Bickel, and T. Scheffer. Unsupervised prediction of citation influences. In *ICML'07*, pages 233–240, 2007.

[9] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed-membership models of scientific publications. *PNAS'04*, 101(Suppl 1):5220–5227, 2004.

[10] D. C. T. Hofmann. The missing link-a probabilistic model of document content and hypertext connectivity. *NIPS'00*, 13:430, 2000.

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

[12] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[13] J. Li, J. Tang, Y. Li, and Q. Luo. Rimom: A dynamic multistrategy ontology alignment framework. *TKDE'09*, 21(8):1218–1232, 2009.

[14] H.-A. Loeliger. An introduction to factor graphs. *Signal Processing Magazine, IEEE*, 21(1):28–41, 2004.

[15] L. Lovász. Random walks on graphs: A survey. *Combinatorics*, 2(1):1–46, 1993.

[16] A. McCallum, A. Corrada-Emmanuel, and X. Wang. Topic and role discovery in social networks. *Computer Science Department Faculty Publication Series*, page 3, 2005.

[17] Q. Mei, D. Cai, D. Zhang, and C. Zhai. Topic modeling with network regularization. In *WWW'08*, pages 101–110, 2008.

[18] D. Mimno, H. M. Wallach, J. Naradowsky, D. A. Smith, and A. McCallum. Polylingual topic models. In *EMNLP'09*, pages 880–889, 2009.

[19] T. Minka. Estimating a dirichlet distribution. Technical report, MIT, 2000.

[20] R. Nallapati, A. Ahmed, E. Xing, and W. Cohen. Joint latent topic models for text and citations. In *KDD'08*, pages 542–550, 2008.

[21] R. Nallapati and W. Cohen. Link-plsa-lda: A new unsupervised model for topics and influence of blogs. In *ICWSM'08*, 2008.

[22] D. Rinser, D. Lange, and F. Naumann. Cross-lingual entity matching and infobox alignment in wikipedia. *Information Systems*, 38(6):887–907, 2013.

[23] 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.

[24] P. Sorg and P. Cimiano. Enriching the crosslingual link structure of wikipedia-a classification-based approach. In *AAAI'08 Workshop on Wikipedia and Artifical Intelligence*, pages 49–54, 2008.

[25] J. Tang, B. Wang, Y. Yang, P. Hu, Y. Zhao, X. Yan, B. Gao, M. Huang, P. Xu, W. Li, and A. K. Usadi. Patentminer: Topic-driven patent analysis and mining. In *KDD'12*, pages 1366–1375, 2012.

[26] J. Tang, S. Wu, J. Sun, and H. Su. Cross-domain collaboration recommendation. In *KDD'12*, pages 1285–1293, 2012.

[27] H. Tong, C. Faloutsos, and J. Pan. Fast random walk with restart and its applications. In *ICDM'06*, pages 613–622, 2006.

[28] M. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

[29] Z. Wang, J. Li, Z. Wang, and J. Tang. Cross-lingual knowledge linking across wiki knowledge bases. In *WWW'12*, pages 459–468, 2012.

[30] J. Winn. Variational message passing and its applications. *Unpublished doctoral dissertation, Cambridge University*, 2003.