Expert Systems With Applications 103 (2018) 106–117

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Hierarchical topic modeling with automatic knowledge mining

Yueshen Xu[a,*], Jianwei Yin[b], Jianbin Huang[a], Yuyu Yin[c]

[a] School of Software, Xidian University, Xi'an, Shaanxi 710071, China
[b] School of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang, 310027, China
[c] Key Laboratory of Complex Systems Modeling and Simulation of Ministry of Education, Zhejiang University, Hangzhou, Zhejiang 310027, China

ARTICLE INFO

Article history:
Received 15 October 2017
Revised 28 January 2018
Accepted 6 March 2018
Available online 7 March 2018

Keywords:
Hierarchical topic modeling
Text mining
Knowledge mining
Non-parametric Bayesian learning
Gibbs sampling

ABSTRACT

Traditional topic modeling has been widely studied and popularly employed in expert systems and information systems. However, traditional topic models cannot discover structural relations among topics, thus losing the chance to explore the data more deeply. Hierarchical topic modeling has the capability of learning topics, as well as discovering the hierarchical topic structure from text data. But purely unsupervised models tend to generate weak topic hierarchies. To solve this problem, we propose a novel knowledge-based hierarchical topic model (KHTM), which can incorporate prior knowledge into topic hierarchy building. A key novelty of this model is that it can mine prior knowledge automatically from the topic hierarchies of multiple domains corpora. In this paper, the knowledge is represented as the word pairs which satisfy the requirement of frequent co-occurrence, and knowledge is organized in form of hierarchical structure. We also propose an iterative learning algorithm. For evaluation, we crawled two new multi-domain datasets and conducted comprehensive experiments. The experimental results show that our algorithm and model can generate more coherent topics, and more reasonable hierarchical structure.

1. Introduction

Traditional topic models, such as LDA (Latent Dirichlet Allocation) (Blei, Ng, & Jordan, 2003) and HDP (Hierarchical Dirichlet Process) (Teh, Jordan, Beal, & Blei, 2006), have been widely used to learn latent topics from text corpora in expert systems, information systems and knowledge management applications (Wu et al., 2017). However, these topic models and their extensions can only find topics in a flat structure, but fail to discover the hierarchical relationship among topics. Such a drawback can limit the application of topic models since many applications need and have inherent hierarchical topic structures, such as the categories hierarchy in Web pages (Ming, Wang, & Chua, 2010), aspects hierarchy in reviews (Kim, Zhang, Chen, Oh, & Liu, 2013) and research topics hierarchy in academia community (Paisley, Wang, Blei, & Jordan, 2015). In a topic hierarchy, the topics near the root have more general semantics, and the topics close to leaves have more specific semantics.

Hierarchical topic modeling is a challenging problem, mainly due to two reasons. The first is that, compared to ordinary topic modeling that only learns topics with no structure, hierarchical topic modeling needs to learn the topics to form meaningful word clusters, as well as the hierarchical structure among topics. The second is that the number of topics at each level is unknown and cannot be set to a predefined value. Some hierarchical topic models (**HTM** for short) have been proposed (Blei, Griffiths, & Jordan, 2010; Blei, Griffiths, Jordan, & Tenenbaum, 2005; Mimno, Li, & McCallum, 2007; Paisley et al., 2015). However, researchers have shown that the existing HTMs often achieve unsatisfactory results, containing incoherent topics and unreasonable structures (Kang, Ma, & Liu, 2012; Mao et al., 2012b). An incoherent topic refers to that in this topic, there exist a part of top topical words that are not consistent with the semantic meaning of other words. The unreasonable structure refers to that in a topic hierarchy, there are some topics in the upper level (parent topics) that cannot be regarded as the generalization of the topics in the associated lower level (children topics). Or some topics in the lower level (children topics) are not the semantic specialization of the topics in the associated upper level (parent topics).

We give a part of topic hierarchy in the following Fig. 1 to illustrate the problems of incoherent topic and unreasonable structure. The shown topic hierarchy is learned by a basic hierarchical topic model, i.e., hLDA (hierarchical LDA), which is proposed by Blei et al. (2010), and is learned from the abstracts corpus of the AAAI Conference on Artificial Intelligence (AAAI for short). The abstracts corpus of AAAI consists of the abstracts of papers published in AAAI over six years, which is crawled by us as one corpus of

[*] Corresponding author.
E-mail addresses: ysxu@xidian.edu.cn, xyshzjucs@zju.edu.cn (Y. Xu), zjuyjw@zju.edu.cn (J. Yin), jbhuang@xidian.edu.cn (J. Huang), yyy718@gmail.com (Y. Yin).

https://doi.org/10.1016/j.eswa.2018.03.008
0957-4174/© 2018 Elsevier Ltd. All rights reserved.

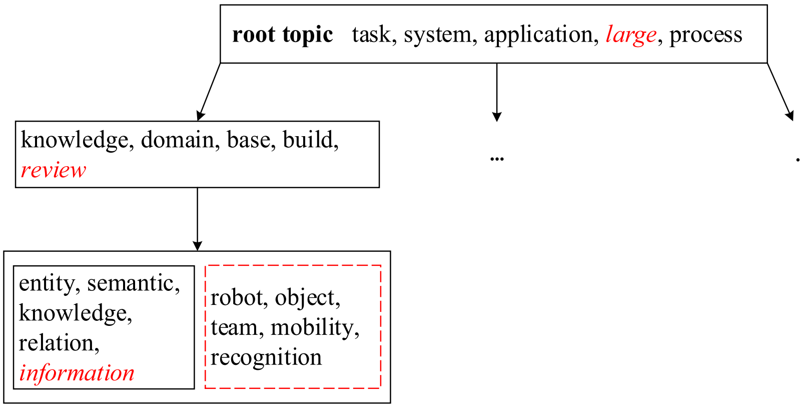

root topic task, system, application, large, process
knowledge, domain, base, build, review
...
...
entity, semantic, knowledge, relation, information
robot, object, team, mobility, recognition

Fig. 1. A part of the topic hierarchy learned from the paper abstracts of AAAI.

the entire multi-domain dataset. The detailed explanation of the datasets used in this paper will be given in the experiment section (Section 5.1).

The shown topic hierarchy is with three level, and it can be observed that there exist several deficiencies, including

1. Incoherence topics. For example, for the root topic with general semantics, the topical word *large* lowers the coherence of the semantic meaning, which originally should be referred to information processing. The same case happens in the shown topic at the second level, the semantic meaning of which originally should be referred to knowledge base. However, the fifth topical word *review* is not consistent with such a semantic meaning, since *review* should have been in a topic on sentiment analysis or text mining.
2. Unreasonable structure. Although the topic at the third level, which is highlighted with red box, has coherent semantic meaning on robotics, this topic should not be a child topic of the topic at the second level. Since the semantic meaning of the topic at the second level is on knowledge base.
3. The third type of deficiencies can be regarded as a special case of unreasonable structure. An example is the fifth topical word *information* in the topic at the third level. That is, as a general word, *information* should be in the topics of upper levels, for example, at the root topic. However, in the current topic hierarchy, *information* is at a specific topic, the semantic meaning of which is on knowledge graph.

Based on the mechanism of hierarchical topic modeling, we find that there are two main reasons leading to the defective topic hierarchies.

1. Similar to ordinary topic modeling, hierarchical topic modeling relies on how often words co-occur in the corpus, i.e., “higher-order co-occurrence”(Heinrich, 2008). So even though some words are semantically more general and should occur in the topics of upper levels, if those words do not occur frequently enough, such words are likely to not occur at the right level. In contrast, if some words are semantically specific but occur frequently, these words are likely to be wrongly at upper levels in the learned topic hierarchy.
2. In hierarchical topic modeling, people also adopt the bag-of-words model, in which words are generated independently, ignoring the semantic relation among words. However, in real-world text, words are generated by certain language rules and probably have semantic relation with each other.

To fix the defects of existing methods, some researchers tried to acquire better topic hierarchies by introducing labeled data (Kang et al., 2012; Mao, He, Yan, & Li, 2012a; Petinot, McKeown, & Thadani, 2011) or linking relationship among documents (Wang, Liu, Desai, Danilevsky, & Han, 2014). However, the labeled data or linking relationships do not exist in every corpus, and probably need much manual work. In this paper, we propose an algorithm to mine prior knowledge from other domain corpora, and use such knowledge to learn a better topic hierarchy. A domain refers to a field, and since we focus on hierarchical topic modeling in text mining, in this paper, a domain is represented by its associated corpus, where the contained documents are related to the same theme or several similar themes. Since the labeled corpora are hard to acquire, our algorithm does not require any labeled data, and does not depend on any linking information. Besides, unlike some other HTMs proposed for specific applications, such as HTM for sentiment analysis (Kim et al., 2013) and HTM for phrase mining (Wang et al., 2013), our algorithm is label and application independent.

There are several existing knowledge-based topic models (Andrzejewski, Zhu, Craven, & Recht, 2011; Chen, Mukherjee, & Liu, 2014; Xie, Yang, & Xing, 2015; Zhang & Zhong, 2016) that can utilize some types of knowledge, for example, the semantic correlation among words. However, their knowledge is provided manually or acquired under flat topics, i.e., with no hierarchical topic structure, so their knowledge cannot be used in our problem. Our proposed algorithm automatically discovers the knowledge, and organizes the knowledge into a hierarchy (see an example in Fig. 3, Section 4.1), which consists of knowledge sets (**k-set** for short). A k-set contains words that are likely to belong to the same topic at a certain level. In this paper, we propose to use k-set as the knowledge unit. We have two observations here.

1. There exist many overlapping topics among different domains. For example, almost every conference with machine learning as the main area or as a sub-area (one conference is regarded as one domain) has topics on supervised learning and unsupervised learning. The topic on supervised learning is likely to be formed by general topical words such as *{training, supervised, class, data}*, and the topic on unsupervised learning is likely to be formed by general topical words such as *{unlabeled, data, clustering, unsupervised}*.
   In this paper, a topic is represented by its top 20 words ranked by probabilities in descending order. The shared topics often contain sets of words frequently co-occurring, which can form potential k-sets.
2. There also exist similar hierarchical structures or sub-structures shared among domains. For example, the two conferences *AAAI* and *NIPS* share the following topic hierarchy on: supervised learning → classification → regression. The topic in the second level on classification is likely to be formed by topical

words such as *{classifier, pattern, label, error}*, and the topic in the third level on regression is likely to be formed by specific topical words such as *{linear, kernel, parameter, regression}*.

In this paper, we use the two types of shared information above as prior knowledge to correct the defective topic hierarchy of each domain corpus. The main contributions of this paper are summarized as follows:

1. It proposes a novel knowledge-based hierarchical topic model (KHTM), which is capable of mining prior knowledge automatically, and incorporating the mined knowledge to learn a superior topic hierarchy. We give the detailed generative process of the model, and the corresponding parameter estimation method based on Gibbs sampling.
2. It proposes a learning algorithm that embeds KHTM in an iterative way, to continuously improve the learning results.
3. It designs a hierarchical structure to maintain knowledge, which is represented as word pairs satisfying frequent co-occurrence with level constraint.
4. It crawled two new multi-domain datasets. Each dataset consists of the paper abstracts of 20 conferences and journals over six years. The first dataset contains abstracts of conferences and journals purely from computer science and the second dataset consists of a mixture of biology, physics and chemistry abstracts. We will release the two datasets publicly to facilitate the related research.

The rest of this paper is organized as follows. Section 2 summarizes the related work. Section 3 states the background model and proposed algorithm. Section 4 elaborates the proposed knowledge-based hierarchial topic model (KHTM). Section 5 presents the experimental results and the analysis. Section 6 concludes the paper.

2. Related work

Hierarchical topic modeling has attracted much attention in expert system community and knowledge management community (Mao et al., 2012a; Pavlinek & Podgorelec, 2017). The task of hierarchical topic modeling is to learn a topic hierarchy from text corpus, in which the topics are organized according to the semantic generality of each topic. A salient characteristic of hierarchical topic modeling is that, for each parent topic, the number of its associated children topics cannot be known or predefined before modeling. So hierarchical topic modeling usually depends on non-parametric Bayesian learning techniques, such as Chinese restaurant process (CRP) or Pachinko allocation.

Blei et al. (2005) used CRP as the non-parametric prior and further proposed the nested Chinese restaurant process (**nCRP**) to achieve hierarchical topic modeling, along with naming the proposed model as hierarchical LDA model (**hLDA** for short). Kim, Kim, Kim, and Oh (2012) proposed the recursive Chinese restaurant process (rCRP) also aiming to achieve hierarchical topic modeling. The difference between nCRP and rCRP is that in rCRP, a document is generated along with multiple paths in the tree-structure of topic hierarchy. By contrast, in nCRP, a document is generated along with one path in the topic hierarchy. Mimno et al. (2007) proposed hierarchical Pachinko allocation model (HPAM) based on Pachinko allocation to generate topic hierarchy. The term Pachinko[1] refers to that in HPAM, a document is generated from a distribution over the topics at the leaves (the lowest level) of a topic hierarchy. The existing hierarchical topic models do not have the capability of mining shared knowledge or using knowledge, so they cannot tackle the problems stated in introduction section, that is, problems arising from higher-order co-occurrence and bag-of-words model.

Some works try to introduce labeled data or linking relationship among documents to improve the quality of topic hierarchy (Mao et al., 2012a,b; Wang et al., 2014). Mao et al. (2012b) assumed that a part of topics in different levels in a topic hierarchy had been known before modeling, and used such known topics as prior knowledge to pre-generate the corresponding structure in a topic hierarchy. We have to note that in many real-world cases, the topics in a hierarchial structure are hard to know beforehand. Even that we know a part of topics, it is also hard to determine the levels of those topics. Wang et al. (2014) used the linking relationship among documents in a heterogeneous network to generate better topic hierarchy. In this paper, we aim to generate topic hierarchy in plain text, not in any specific environment, such as heterogeneous network.

Our work is also related to knowledge-based topic models, since our idea contains the part to learn shared knowledge and use the learned knowledge in modeling. In many knowledge-based topic models, the used knowledge, such as the lexical relation or semantic relation, needs to be collected manually (Andrzejewski et al., 2011; Jagarlamudi, Daumé, & Udupa, 2012; Mukherjee & Liu, 2012). In our model, the used knowledge is mined automatically without any manual work. There are also some topic models that can use knowledge beyond the corpus (Chen et al., 2014; Wang, Chen, & Liu, 2016). However, their task is traditional topic modeling, which learns flat topics without any structure. So their used knowledge has no hierarchical structure either. In this paper, our goal is to learn hierarchical topics, and our learned knowledge also has hierarchical structure (An example will be given in Section 4.1).

3. Base model and the proposed learning algorithm

In this paper, we employ the hierarchical LDA model (**hLDA** for short) (Blei et al., 2005) as base model. In this section, we first briefly review the hLDA model, and then present our proposed algorithm.

3.1. Hierarchical LDA

hLDA learns and organizes topics into an L-level tree-like hierarchy corresponding to the semantic generality of topics. hLDA is based on a non-parametric prior, i.e., nested Chinese restaurant process (nCRP), and nCRP is extended from the Chinese restaurant process (CRP).

For CRP, let us imagine a restaurant with an infinite number of tables, and customers enter this restaurant sequentially. The mth customer sits at a table (c_m) according to the probability drawn from the following distribution:

$$p(c_m = l | c_1, c_2, \ldots, c_{m-1}) \propto \begin{cases} n_l, & \text{if } l \text{ is previous occupied} \\ \gamma, & \text{if } l \text{ is a new table} \end{cases} \tag{1}$$

where n_l is the number of customers who have sat at table l before, and γ is a hyperparameter.

For nCRP, let us imagine the following hierarchical structure: at the first level, there is one restaurant with only one table, which is linked with an infinite number of tables at the second level. Each table at the second level is also linked with an infinite number of tables at the third level. Such a structure is repeated until the Lth level. Each customer starts at the first level, sits at only one table at each level, and ends at the Lth level, which produces an L length path along with siting at each table. The selection of a table follows the CRP prior (see Eq. (1)). In topic modeling, a table refers to a topic, and a customer refers to a word.

[1] The original meaning of Pachinko refers to a pinball game in Japan.

Table 1
The domain list: Computer Science dataset (1st row) and Natural Science dataset (2nd row).

Domain names (conference or journal): AAAI, ACM Multimedia, CIKM, CVPR, ECML, ICCV, ICDE, ICDM, ICML, IJCAI, JMLR, NIPS, SAC, SDM, SIGIR, SIGKDD, SIGMOD, SIGSpatial, VLDB, WWW
Domain names (conference or journal): ACM BCB, Bioinformatics, IEEE TCBB, Journal of Computational Biology, Molecular Biology of the Cell, Nature Reviews : Molecular Cell Biology, Nucleic Acids Research, Journal of American Chemical Society, Chemistry Central Journal, Chemistry-A European Journal, Journal of Biological Chemistry, Journal of Chemical Physics, Journal of Physical Chemistry: A, Nature Chemical Biology, Applied Physical Letter, Biophysical Journal, Modern Physics Letter: B, Physical Review Letters, Physical Chemistry Letter, Progress of Theoretical and Experimental Physics (PTEP)

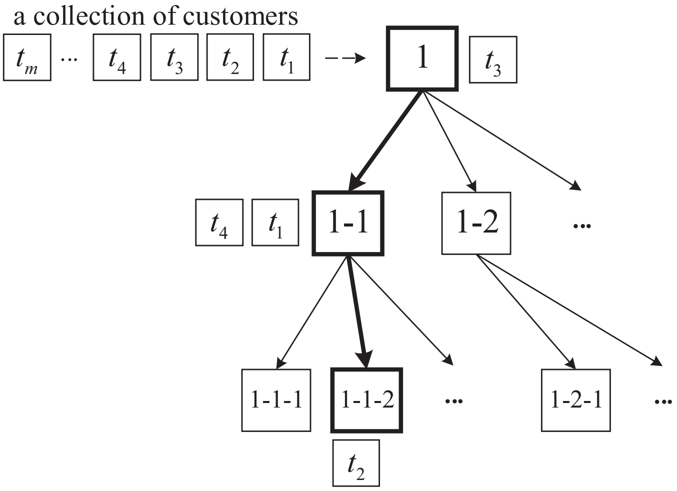


Fig. 2. An example to illustrate the process of nCRP.

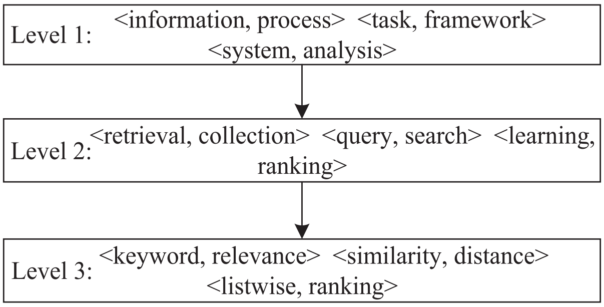

Level 1:
<information, process> <task, framework>
<system, analysis>
Level 2:
<retrieval, collection> <query, search> <learning, ranking>
Level 3:
<keyword, relevance> <similarity, distance>
<listwise, ranking>

Fig. 3. A part of the mined knowledge hierarchy (F) over SIGIR domain corpus.

To further explain how nCRP generates a hierarchical structure, we present a figure to illustrate the process of nCRP, which is shown in the following Fig. 2. In Fig. 2, we have a collection of customers (t_1, t_2, t_3, ..., t_m), queuing sequentially. At the first level, there is only one restaurant contained only one table (notated as Table 1). Table 1 is linked with an infinite number of tables (notated as Table 1-1st row, Table 1-2nd row, ...) at the second level. Then for the third level, take Table 1-1st row as an example. Table 1-1st row is also linked with an infinite number of tables (notated as Table 1-1st and 1st row, Table 1-1 and 2nd row, ...) at the second level. In the customer side, customer t_3 sits at Table 1, t_1 and t_4 sit at Table 1-1st row, and t_2 sits at Table 1-1st and 2nd row. Such a way of customer sitting forms a hierarchical structure.

3.2. The proposed learning algorithm

Our proposed algorithm consists of three steps.

Step 1 (Preliminary topic hierarchy learning): Given a set of domains $D = \{D_i\}_{i=1}^{I}$, where I is the number of domains, our proposed algorithm (Algorithm 1) first runs hLDA to generate a preliminary topic hierarchy (T_i) for each domain D_i (lines 3–5). Then the topic hierarchies from all domains are combined to produce the union set T (line 9), which is used to generate prior knowledge in our proposed model KHTM (Algorithm 2) in the next iteration.

Algorithm 1: The proposed learning algorithm(D).

```
1:  for $h = 1$ to $H$ do
2:      for each domain $D_i \in D$ do
3:          if $h = 1$ then
4:              $T_i \leftarrow$ hLDA($D_i$)
5:          else
6:              $T_i \leftarrow$ KHTM($D_i$, $T_i$, $T$)
7:          end if
8:      end for
9:      $T \leftarrow \cup_i T_i$
10: end for
```

Algorithm 2: KHTM(D_t, T_t, T).

```
for each topic $a \in T_t$ do
    for each topic $a' \in T$ do
        if $level(a') = level(a) \wedge$ Jaccard $(a', a) > \rho$ then
            $S_a \leftarrow S_a \cup \{a'\}$
        end if
    end for
end for
$F \leftarrow \cup_a$ Frequent-Item-Mining $(S_a)$ // Details in Section 4.1
$T_t^{new} \leftarrow$ GibbsSampling$(D_t, F)$    // Details in Section 4.3
return $T_t^{new}$
```

Step 2 (Iterative learning): After the first iteration, KHTM is called to first mine k-sets from the union topic hierarchies set T, and then incorporate the mined k-sets to produce a better topic hierarchy for each domain D_i (line 6). In Algorithm 1, H is the number of iterations. We will study the convergence of Algorithm 1 and the setting of H in the experiment section (Section 5.3).

Step 3 (Target topic hierarchy learning): Given a target domain D_t and its corpus, and a topic hierarchies set T, KHTM is called to generate the topic hierarchy for D_t. The KHTM model will be elaborated in Section 4. D_t can be a domain in D (i.e., $D_t \in D$) or a totally new domain (i.e., $D_t \notin D$). In the case of $D_t \in D$, D_t represents each domain D_i ($i = 1, \ldots, I$). The target topic hierarchy is the topic hierarchy that is learned at the last iteration for each domain. In the case of $D_t \notin D$, our algorithm mines k-sets from the previous learned topic hierarchies set, and just applies the resulting knowledge in modeling the new domain D_t. We will study the performance of KHTM under the two settings in the experiment section.

4. The proposed KHTM model

This section elaborates the proposed model KHTM (Algorithm 2), which consists of three components: knowledge mining, knowledge utilization and parameter estimation.

4.1. Knowledge mining

Knowledge mining (lines 1–8, Algorithm 2) is conducted as follows.

Topic filtering (lines 1–7, Algorithm 2): In each domain D_t that is currently being modeled, we select similar topics for each topic $a \in T_t$ from the topic hierarchies set T under two constraints (line 3): 1) the existing topic $a' \in T$ should be at the same level as a. 2) a' is similar to a measured by Jaccard coefficient. ρ is a similarity threshold to filter topics dissimilar to a. The selected topics form the similar topics set S_a of topic a (line 4).

Knowledge hierarchy building (line 8): Given the similar topics set S_a, this substep first uncovers sets of words that co-occur frequently in S_a using frequent itemset mining technique (Agrawal & Srikant, 1994). Each topic is regarded as a transaction consisting of top 20 topical words. A word set (i.e., frequent itemset) is a knowledge set (k-set), and has an attribute *level* specifying the topic level where it is mined from. A k-set is restricted to contain two words, since a pairwise relation has sufficient capability to reflect the topic co-occurrence among words. The k-sets of all topics $a \in T_t$ are organized into a knowledge hierarchy (represented by F in line 8) according to the topic level. We give a part of the mined knowledge hierarchy in Fig. 3, which is mined on the SIGIR abstracts. SIGIR is regarded as a domain in Computer Science dataset, and the paper abstracts of SIGIR conference over six years form the domain corpus of SIGIR.

We take the k-set $<$*similarity, distance*$>$ at the third level as an example to explain. Here the SIGIR domain is D_t. The k-set $<$*similarity, distance*$>$ indicates that the two words *similarity* and *distance* frequently co-occur in the topics at the third level of the topic hierarchies of many domain corpora, for example, the abstracts corpora of CIKM,[2] SIGKDD[3] and ICDM.[4] Meanwhile, those topics containing *similarity* and *distance* from other domain corpora, are similar to a topic ($a \in T_t$) that is at the third level of the topic hierarchy of SIGIR domain corpus. Using frequent itemset mining technique, the word pair $<$*similarity, distance*$>$ is mined as a frequent k-set from one topics set S_a, and all mined k-sets form F according to their associated levels, a part of which is shown in Fig. 3.

4.2. Knowledge utilization

An intractable problem is how to utilize the knowledge hierarchy in hierarchical topic modeling. Since a k-set illustrates a frequent topic co-occurrence of two words, our idea is to increase the probability of the two words occurring in the same topic. We achieve this goal by employing *generalized Pólya urn* model (GPU) (Mimno, Wallach, Talley, Leenders, & McCallum, 2011).

The GPU model is extended from the *simple Pólya urn* (SPU) model, which employs colored balls and urns to depict the sampling process. In topic modeling, a word denotes a ball with a color and a topic denotes an urn. In SPU, after a ball is drawn from an urn, this ball is put back into the urn with a new ball of the same color. The traditional topic models, such as LDA and hLDA, follow the SPU model. In GPU, after two balls with the same color are put into the urn, a certain quantity of some other balls with multiple colors will be also put into the urn. Such promotion increases the proportion of the balls with other colors.

We propose the following rule of knowledge utilization based on the sampling process of GPU. In the KHTM model, after the word w' is assigned to a topic at the lth level, word w will be promoted with being assigned to that topic by a certain quantity. This quantity is determined by tensor $\Lambda = \{\lambda_{l,w',w}\} \in \mathbb{R}^{L\times V\times V}$ under two constraints:

1. $\{w', w\}$ must be a k-set.
2. *level*($\{w', w\}$), the level of the k-set, must be equal to the level l of the assigned topic.

Otherwise, $\lambda_{l,w',w}$ is equal to 0. The promotion value in $\Lambda = \{\lambda_{l,w',w}\}$ should be proportional to the correlation of w' and w in the target domain corpus. The problem next is how to set the value of $\lambda_{l,w',w}$. In this paper, we utilize PMI (Pointwise Mutual Information) to measure the correlation of two words (Newman, Lau, Grieser, & Baldwin, 2010). In this case, PMI is the logarithm rate between the joint probability of two words co-occurring and the probability of the two words occurring independently. PMI measures the extent to which two words are likely to co-occur. That is, a larger PMI value means a higher-order co-occurrence, and a smaller PMI value means a lower-order co-occurrence. The PMI value of two words w' and w is computed as

$$PMI(w', w) = \log \frac{p(w', w)}{p(w')p(w)} \tag{2}$$

where $p(w')$ and $p(w)$ are the document frequency ratios of w' and w in the domain corpus respectively, and $p(w', w)$ is the document frequency ratio of words w' and w co-occurring in the domain corpus. $p(w')$ and $p(w)$ indicate the possibility of seeing w' and w in a random document, and $p(w', w)$ indicates the possibility of seeing both w' and w in a random document.

We further introduce a parameter ξ to control the extent of trusting the PMI value, and the setting of ξ will be given in Section 5.1, and we will also give the sensitivity of our model to ξ in the experiment section. We use PMI and ξ to provide an automatic way to handle wrong knowledge in k-sets. That is, a frequent k-set mined from other domains does not necessarily mean that the words in this k-set are coherently associated in the target domain, mainly due to the polysemy issue. So finally, the promotion quantity for w when sampling a topic for w' is set as follows.[5]

$$\lambda_{l,w',w} = \begin{cases} \xi \times PMI(w', w), & \{w', w\} \text{ satisfies the two constraints}^{1} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

4.3. Generative process and parameter estimation

This subsection presents the generative process and parameter estimation of the KHTM model. Similar to hLDA, the generative process of KHTM model is also based on nest Chinese Restaurant Process (nCRP), and the detail of nCRP has been stated in Section 3.1, so we focus on KHTM model here. Let $\mathbf{w}_m$ represent the words of document m. A document m is generated by first sampling an L-level path through the topic hierarchy formed by the restaurants, and then sampling each word $w_{m,\,n}$ ($w_{m,\,n} \in \mathbf{w}_m$) from the L topics that correspond to the restaurants along the sampled path. Compared to hLDA, one salient difference is that in KHTM, after a word is sampled and assigned to a certain topic, some other

[2] CIKM is short for International Conference on Information and Knowledge Management.

[3] SIGKDD is short for ACM Conference on Knowledge Discovery and Data Mining.

[4] ICDM is short for International Conference on Data Mining.

[5] $\{w', w\}$ is a k-set and the level of $\{w', w\}$ is equal to the level of the current sampled topic.

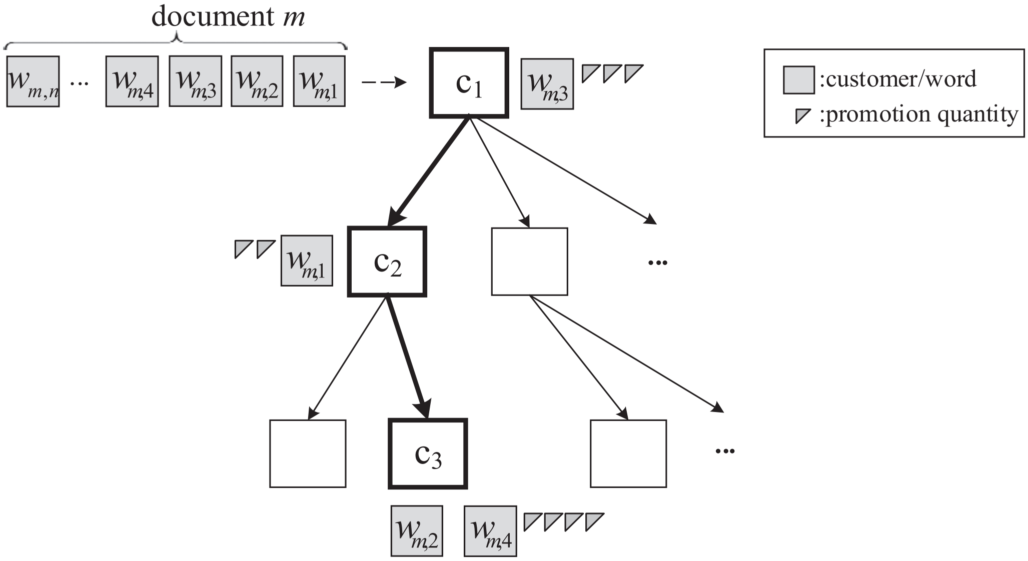


Fig. 4. An example to illustrate the generative process of KHTM model.

related words will be also assigned to that topic with the promotion quantities defined in Λ (see Eq. (3)). In detail, for each document m, the generative process of KHTM model is as follows.

1. Let c_1 be the shared root restaurant.
2. For each level $l \in \{2, \ldots, L\}$,
   (a) Sample a table c_l from restaurant c_{l-1} according to Eq. (1). c_l is also the restaurant referred to by the sampled table.
3. Sample a topic proportion vector $\boldsymbol{\theta}_m$ with L-dimension from $Dir(\alpha)$.
4. For each word $w_{m,\,n} \in \mathbf{w}_m$,
   (a) Sample $l \in \{1, \ldots, L\}$ from $Mult(\boldsymbol{\theta}_m)$.
   (b) Sample $w_{m,\,n}$ from the topic associated with restaurant c_l.
   (c) Sample other words with the promotion quantity in tensor Λ according to Eq. (3).

In the generative process, $Dir(\alpha)$ is an L-dimensional Dirichlet distribution with α as the hyperparameter, and $Mult(\boldsymbol{\theta}_m)$ is the multinomial distribution with the topic proportion vector $\boldsymbol{\theta}_m$ as the variable.

We give an example in Fig. 4 to illustrate the generative process of KHTM model, where the total level is 3. First, according to the second step in the above proposed generative process, in the current topic hierarchy, document m is sampled to a path, which consists of three tables/topics, i.e., $c_1 \rightarrow c_2 \rightarrow c_3$, where c_1 is the root general topic, c_2 is a topic at the second level, and c_3 is a specific topic at the third level. Then, the words in document m are sequentially sampled to a topic from the three topics c_1, c_2 and c_3. As the example shown in Fig. 4, $w_{m,\,3}$ is sampled to c_1 probably because $w_{m,\,3}$ is a word with general semantics. $w_{m,\,4}$ is sampled to c_3 probably because $w_{m,\,4}$ is a word with specific semantics. After a word is sampled to a topic, the other words associated with it will also be sampled into that topic with the promotion quantity, which is computed as the value of $\lambda_{l,w',w}$ in tensor Λ (see Eq. (3)). In Fig. 4, the promotion quantity ($\lambda_{l,w',w}$) is represented as a small triangle, since the value of $\lambda_{l,w',w}$ is usually a decimal, representing a fraction of the promoted word. Note that, in Fig. 4, there is no promotion quantity associated with $w_{m,\,2}$, which means that there are no words promoted by $w_{m,\,2}$ at the third level. It further means that no k-sets containing $w_{m,\,2}$ at the third level can be applied in the current domain corpus.

Gibbs sampling. We utilize Gibbs sampling (Griffiths & Steyvers, 2004) to estimate the parameters in KHTM. Corresponding to the generative process, there are two necessary steps in Gibbs sampling.

1. Sampling a path assignment $\mathbf{c}_m = \{c_{m,l}\}_{l=1}^{L}$ for each document m in the domain corpus.
2. Sampling a level l (a topic $z_{m,\,n}$) along the path for $w_{m,\,n}$, the nth word in document m.

For the first step, given the assignment of $z_{m,\,n}$, the conditional distribution of $\mathbf{c}_m$ is

$$p(\mathbf{c}_m|\mathbf{w}, \mathbf{c}_{-m}, \mathbf{z}) \propto \\ p(\mathbf{w}_m|\mathbf{c}, \mathbf{w}_{-m}, \mathbf{z}) \times p(\mathbf{c}_m|\mathbf{c}_{-m}) \tag{4}$$

where $\mathbf{w}$ denotes all words in the domain corpus, $\mathbf{c}$ denotes the path assignments of all documents, and $\mathbf{z}$ denotes the topic assignments of all words. $\mathbf{w}_{-m}$ represents all words in the domain corpus except the words in document m. $\mathbf{c}_{-m}$ represents the path assignments of all documents except document m. $p(\mathbf{c}_m|\mathbf{c}_{-m})$ is the prior following nCRP (Section 3.1). As the likelihood, $p(\mathbf{w}_m|\mathbf{c}, \mathbf{w}_{-m}, \mathbf{z})$ is computed with

$$\begin{aligned} & p(\mathbf{w}_m|\mathbf{c}, \mathbf{w}_{-m}, \mathbf{z}) \\ & = \prod_{l=1}^{L} \left(\frac{\Gamma(n^{(\cdot)}_{c_{m,l},-m,-\sum_m} + |V|\eta)}{\prod_w \Gamma(n^{(w)}_{c_{m,l},-m,-\sum_m} + \eta)} \right. \\ & \quad \left. \times \frac{\prod_w \Gamma(n^{(w)}_{c_{m,l},-m,-\sum_m} + n^{(w)}_{c_{m,l},m,\sum_m} + \eta)}{\Gamma(n^{(\cdot)}_{c_{m,l},-m,-\sum_m} + n^{(\cdot)}_{c_{m,l},m,\sum_m} + |V|\eta)} \right) \end{aligned} \tag{5}$$

where $n^{(w)}_{c_{m,l},-m,-\sum_m}$ is the number of instances of word w assigned to the topic indexed by $c_{m,\,l}$, excluding those in document m and Σ_m. $\sum_m = \{\sum_{w' \in m} \lambda_{l,w',w}\}_{w \in V}$ represents the collection of additional values promoted by the words in document m. Specially, $\lambda_{l,w',w}$ is the promotion value of w' to w at level l. Similarly, $n^{(\cdot)}_{c_{m,l},-m,-\sum_m}$ denotes the number of words assigned to the topic indexed by $c_{m,\,l}$, excluding those in document m and Σ_m. $|V|$ is the vocabulary size. $\Gamma(\cdot)$ is the standard gamma function.

For the second step, given the current state of Gibbs sampler, we sample $z_{m,\,n}$ for word $w_{m,\,n}$. Let us set an index $i = (m, n)$ referring to the word $w_{m,\,n}$. As an variable, the topic proportion vector $\boldsymbol{\theta}_m$ is integrated out, and we can get the conditional probability of

Table 2
Statistics of Computer Science dataset.

	AAAI	ACM MM	CIKM	CVPR	ECML	ICCV	ICDE	ICDM
#Doc	2247	1682	2120	2724	578	557	691	663
#Avg	142	145	159	157	145	151	175	172
	ICML	IJCAI	JMLR	NIPS	SIGKDD	SAC	SDM	SIGIR
#Doc	1277	2322	614	1998	1055	2138	550	1291
#Avg	132	125	144	142	187	134	173	151
	SIGMOD	SigSpatial	VLDB	WWW				
#Doc	806	525	1073	1013				
#Avg	176	153	173	159				

Table 3
Statistics of Natural Science dataset.

	ACM BCB	Bioinformatics	IEEE TCBB
#Doc	626	2804	645
#Avg	178	153	173
	JCB	Molecular Biology of the Cell	Molecular Cell Biology
#Doc	942	1095	276
#Avg	181	182	99
	Nucleic Acids Research	JACS	Chemistry Central Journal
#Doc	3676	1003	455
#Avg	182	128	212
	Chemistry-A European Journal	Journal of Biological Chemistry	Journal of Chemical Physics
#Doc	1983	2696	1707
#Avg	52	206	148
	Journal of Physical Chemistry:A	Nature Chemical Biology	Applied Physical Letter
#Doc	1141	484	1544
#Avg	168	120	89
	Biophysical Journal	Modern Physics Letter:B	Physical Review Letters
#Doc	2334	913	1095
#Avg	80	116	106
	Physical Chemistry Letter	PTEP	
#Doc	582	602	
#Avg	163	125	

assigning the topic at the lth level to w_i as follows.

$$p(z_i = l|\mathbf{z}^{-i}, \mathbf{w}, \mathbf{c}_m, \Lambda) \propto \frac{n^{-i}_{m,l} + \alpha}{\sum_{l'=1}^{L}(n^{-i}_{m,l'} + \alpha)} \times \frac{\sum_{w'=1}^{V} \lambda_{l,w',w_i} \times n^{-i}_{l,w'} + \eta}{\sum_{v=1}^{V}(\sum_{w'=1}^{V} \lambda_{l,w',v} \times n^{-i}_{l,w'} + \eta)} \tag{6}$$

where $\mathbf{z}^{-i}$ denotes the topic assignment of all words except word w_i ($w_{m,\,n}$). $n^{-i}_{m,l}$ is the number of words in document m assigned to topic l and $n^{-i}_{l,w'}$ is the number of word w' assigned to topic l, both except the word w_i. α and η are two hyperparameters. Λ is the tensor of promotion quantities. Using Eq. (6), we will know the topic assignment at a certain level of each word in the corpus, and then we can construct the topic hierarchy.

5. Experiment and evaluation

5.1. Experiment setting

Datasets. We crawled two new datasets for the experiments. The first dataset contains the paper abstracts of 20 conferences and journals from computer science over six years, including AAAI, CIKM, CVPR, SIGIR, etc. We name this dataset *Computer Science* dataset. The second dataset contains the paper abstracts of 20 conferences and journals from biology, chemistry and physics, also over six years. We name this dataset *Natural Science* dataset. In both datasets, each conference or journal is regarded as a domain, and each abstract is regarded as a document. The complete domain lists of the two datasets are shown in Table 1.[6] The detailed information of each domain, including the number of documents and average length of a document, is given in Table 2 (Computer Science dataset) and Table 3 (Natural Science dataset). In the two tables, *#Doc* denotes the number of documents in a domain corpus, and *#Avg* denotes the average length of the documents in a domain.

Using the two datasets, we aim to evaluate our proposed model in two settings: the one with large topic hierarchies overlapping (Computer Science dataset) and the other one with limited topic hierarchies overlapping (Natural Science dataset). In each domain corpus, we removed the stop words. Also, we removed those too frequent words, which are measured by document frequency (larger than 40%).

Baseline models. We evaluate the performance of our KHTM model compared to the following baseline models.

hLDA: the hierarchical LDA model proposed in Blei et al. (2005).

hLDA_Union: We union the abstracts of all conferences and journals together into one single corpus, and run hLDA on this corpus. We notate the hLDA model in this setting as *hLDA_Union*.

[6] In Table 1, ACM BCB is short for ACM Conference on Bioinformatics, Computational Biology, and Health Informatics. IEEE TCBB is short for IEEE Transactions on Computational Biology and Bioinformatics. PTEP is short for Progress of Theoretical and Experimental Physics.

HPAM: the hierarchical Pachinko allocation model (HPAM) proposed in Mimno et al. (2007), which also aims to generate topic hierarchy.

KHTM(A): a variant of KHTM, which does not use the level constraint (Algorithm 2, line 3) when the model tries to filter dissimilar topics. That is, all topics at all levels from other domains can be the similar topics of a topic in the current domain.

Parameter setting. For the Dirichlet hyperparameters α and η, we set $\alpha = 1$ and $\eta = 0.1$ symmetrically. The hyperparameter λ in CRP (see Eq. (1)) is set to 0.1 following Blei et al. (2010). The similarity threshold ρ in KHTM model (line 3) is set to 0.2. A large ρ gives a strict threshold to filter dissimilar topics. A small ρ allows more existing topics to be similar topics. Our preliminary experiments found that the empirical value $\rho = 0.2$ produces satisfactory topic hierarchy results. The parameter ξ in the GPU model is also set to 0.2, which controls the extent of trusting the promotion value of words in a k-set. We will study the sensitivity of our model to ξ in Section 5.4. All other parameters in baseline models are set according to their original papers.

5.2. Topic Coherence Comparison

In this paper, we use the measure Topic Coherence, rather than perplexity, to evaluate the quality of topics in a topic hierarchy. In Chang, Boyd-Graber, Gerrish, Wang, and Blei (2009), the authors verified that perplexity does not always conform with human judgements, and cannot correctly measure whether a topic is semantically coherent or not. By comparison, Mimno et al. (2011) showed that Topic Coherence is consistent with human interpretation. Topic Coherence has been used popularly by other researchers (Wang et al., 2016; Yan, Guo, Lan, & Cheng, 2013). Topic Coherence is computed with

$$C(T^z) = \sum_{k=2}^{K} \sum_{j=1}^{k-1} \log \frac{D(t_k^z, t_j^z) + 1}{D(t_j^z)} \tag{7}$$

where $T^z = (t_1^z, t_2^z, \ldots, t_K^z)$ represents topic z, consisting of the top K words with the largest probabilities. $D(t_k^z, t_j^z)$ is the document frequency of the two topical words t_k^z and t_j^z co-occurring in corpus. $D(t_j^z)$ is the document frequency of t_j^z in corpus. As a way of Laplacian smoothing, in the numerator, $D(t_k^z, t_j^z)$ is added by 1 to avoid $D(t_k^z, t_j^z)$ to be 0. A higher Topic Coherence means a higher quality topic.

Test Setting. We evaluate the KHTM model under the following two settings, as stated in Section 3.2.

Test setting 1: the target domain D_t is contained in the collection of domains D, i.e., $D_t \in D$. We mark this setting as *KHTM(S1)*.

Test setting 2: the target domain D_t is a new domain, i.e., $D_t \notin D$. In this setting, KHTM generates the topic hierarchy for D_t using the knowledge mined from the saved topic hierarchies set of existing domains D. We mark this setting as *KHTM(S2)*.

In the two settings, each domain is used to be the target domain D_t and the rest of domains form the collection of domains D. All models are compared by the average Topic Coherence over all target domains. The experimental results are given in Figs. 5 and 6, in which the total hierarchical level is 3.

From Figs. 5 and 6, we can make four observations as follows.

1. The proposed model KHTM(S1) achieves the highest Topic Coherence in both datasets, and KHTM(S2) also achieves satisfactory performance. It indicates that the knowledge (k-set) mined from other domains can indeed help learn a better topic hierarchy in the target domain. It also verifies the effectiveness of the designed Gibbs sampling algorithm, to effectively estimate the parameters. The performance of KHTM(S2) is slightly worse than that of KHTM(S1). The reason is that in test setting 2, the target domain D_t does not join knowledge mining in the topic hierarchies set of D, which lowers the quality of knowledge, and subsequently lowers the topic quality of D_t.
2. KHTM(A) achieves better performance than hLDA, but is poorer than KHTM(S1) and KHTM(S2). It indicates that in hierarchical topic modeling, mining k-sets without level constraint is harmful to topic quality. The prior knowledge should be mined specific to a certain level.
3. hLDA_Union performs worse than hLDA. We also manually checked the results of hLDA_Union, and found that the learned topic hierarchy contains many incoherent topics. It illustrates that it is hard to gain a good topic hierarchy if we run a hierarchial topic model on a large corpus containing too many domains. A more effective way is to use multi-domain knowledge mined from the result of each domain, as KHTM does.
4. The improvement achieved by KHTM(S1) and KHTM(S2) is less in Natural Science dataset. That is expected since the domains in this dataset are from different disciplines (biology, chemistry and physics), so that they do not share as much knowledge (k-set) as in Computer Science dataset. But even in Natural Science dataset, our model still outperforms baseline models.

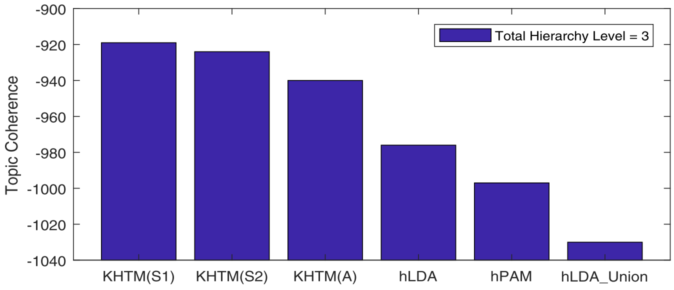


Fig. 5. Topic Coherence Comparison (Computer Science dataset).

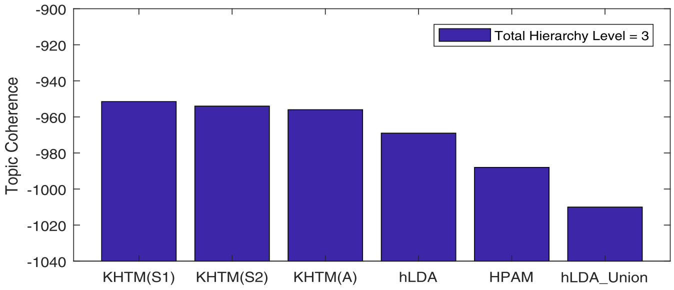


Fig. 6. Topic Coherence Comparison (Natural Science dataset).

We also show the Topic Coherence Comparison of each domain in Computer Science dataset in Table 4.[7] The total hierarchical level is 3. It can be seen that the KHTM(S1) model achieves the best performance in all domains. In both datasets, the improvements achieved by KHTM(S1) are significant over baseline models based on paired t-test ($p < 0.001$) over all 20 domains. We also conducted the experiments under the case that the total hierarchy level is 4. We observed that the KHTM model also achieves significant improvements.

5.3. The convergence of Algorithm 1

In this section, we study the convergence of Algorithm 1 and the setting of the iteration number H. As a core algorithm, Algorithm 1 is given in Section 3. The experimental results are shown in Fig. 7 (Computer Science dataset) and Fig. 8 (Natural Science dataset). The total hierarchical level is 3 and 4.

[7] In Table 4, ACM MM is short for ACM Multimedia Conference.

Table 4
Topic Coherence Comparison in Computer Science dataset (a larger value means a better performance).

	AAAI	ACM MM	CIKM	CVPR	ECML	ICCV	ICDE
hLDA	−1083	−1042	−1101	−1129	−875	−874	−905
KHTM(A)	−1033	−988	−1063	−1040	−862	−873	−865
KHTM(S1)	−931	−984	−1038	−1027	−835	−845	−862
	ICDM	ICML	IJCAI	JMLR	SIGKDD	NIPS	SAC
hLDA	−906	−916	−1082	−863	−974	−1094	−1092
KHTM(A)	−898	−838	−1021	−839	−985	−1014	−1071
KHTM(S1)	−858	−836	−970	−835	−955	−955	−954
	SDM	SIGIR	SIGMOD	SIGSpatial	VLDB	WWW	
hLDA	−858	−1012	−928	−814	−1003	−966	
KHTM(A)	−845	−979	−923	−822	−962	−921	
KHTM(S1)	−843	−977	−911	−806	−956	−875	

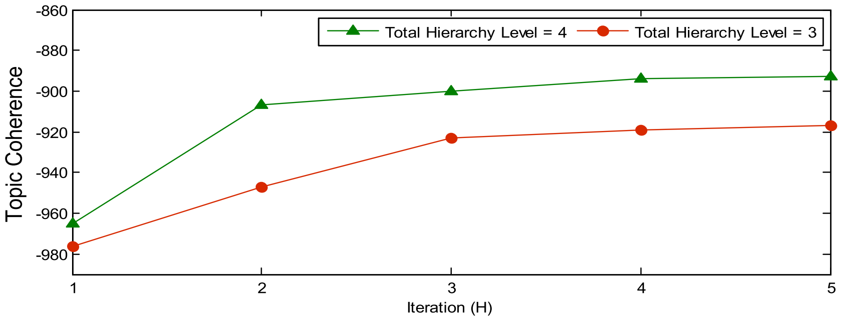


Fig. 7. The Convergence of Topic Coherence (Computer Science dataset).

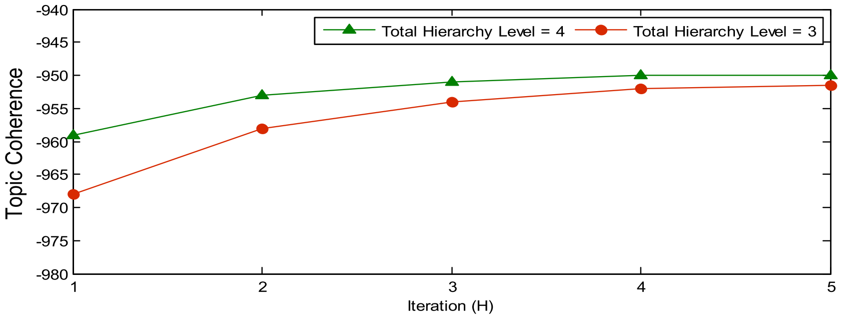


Fig. 8. The Convergence of Topic Coherence (Natural Science dataset).

From Figs. 7 and 8, it can be seen that the proposed algorithm is converged after four iterations. Thus, we can get a stable result after only several iterations. We can observe that the Topic Coherence values are continuously improved along with the iterations increasing. From such continuous improvement, it can be also inferred that the quality of knowledge, which is formed by k-sets and organized in a hierarchy (see Fig. 3), is improved with the iteration increasing. It also indicates the effectiveness of the design of knowledge hierarchy, which is formed by k-sets.

5.4. The sensitivity to ξ

The parameter ξ controls the extent of word association in a k-set measured by PMI. A larger ξ means a higher confidence to the word association in the target domain. In contrast, a smaller ξ means a lower confidence. We studied the sensitivity of the proposed model KHTM (in test setting 1) to ξ in Computer Science dataset. We report the experimental result of the total hierarchical level being 3 (shown in Fig. 9).

Fig. 9 shows that the KHTM model achieves the highest performance in the range of 0.1 to 0.4. It illustrates that our model gives enough room to choose a suitable value (0.1 $\sim$ 0.4) for ξ to achieve a superior topic hierarchy. Also, it can be inferred that the utilization of ξ is necessary, since after $\xi > 0.4$, the performance becomes worse. However, even the worst performance of KHTM, which is achieved at $\xi = 1$ (i.e., we totally trust the PMI value), is still better than the performance of hLDA. The result of hLDA keeps the same (i.e., -976) in the whole range since it does not use any knowledge. We also observed the similar performance sensitivity of our model to ξ in test setting 2.

5.5. Hierarchical structure analysis

Besides coherent topics, an effective hierarchical topic model should also produce a topic hierarchy with a reasonable structure. That is, the topics near the root should have more general semantics, and the topics near the leaves should have more specific semantics. In this paper, we adopt the metric Topic Specialization (**TS** for short) proposed in Kim et al. (2012) to evaluate the structure quality. The following is the intuition and computation of Topic Specialization. First, we need to define a general topic ϕ_{norm}. In ϕ_{norm}, the occurrence probability of a word w_i is proportional to the document frequency of w_i. Let $D(w_i)$ be the document frequency of w_i in corpus, V be the whole unique vocabulary, and

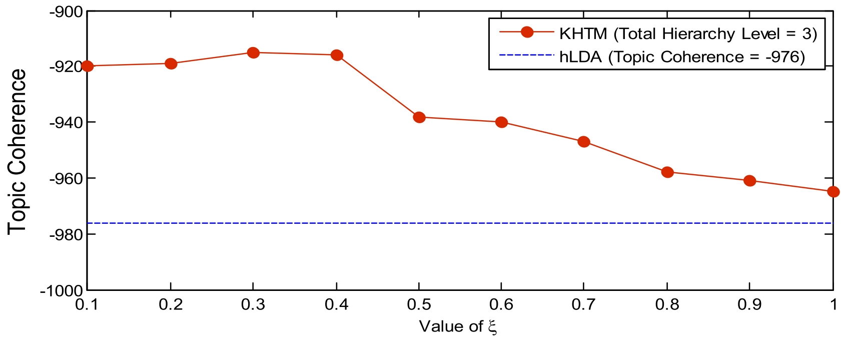
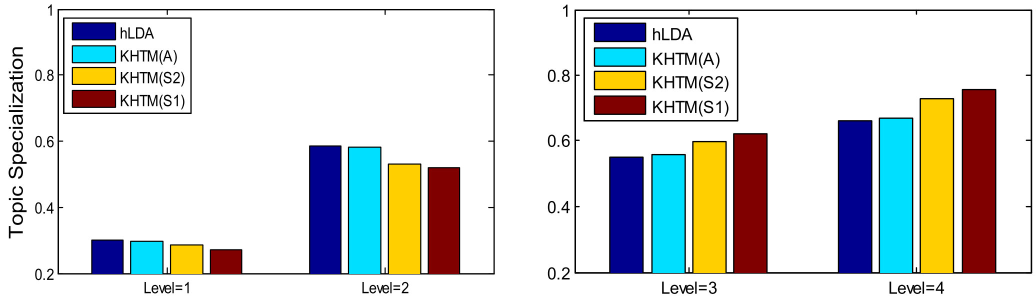


Fig. 9. The Sensitivity to ξ (Computer Science dataset).



(a) Hierarchy Level

(b) Hierarchy Level

Fig. 10. Topic Specialization Comparison (Computer Science dataset).

δ be the smoothing factor avoiding $D(w_i)$ to be 0. The occurrence probability of w_i in the general topic ϕ_{norm} is

$$p(w_i|\phi_{norm}) = \frac{D(w_i) + \delta}{\sum_{j\in V} D(w_j) + \delta|V|} \tag{8}$$

Since ϕ_{norm} represents the word distribution over the whole corpus, ϕ_{norm} can be regarded as the topic with the most general semantics. For each topic ϕ_k in the topic hierarchy, we measure the topic specificity by computing the deviation extent of ϕ_k to ϕ_{norm}, which is further computed by the cosine similarity between ϕ_k and ϕ_{norm}. The deviation has an inverse correlation with cosine similarity. That is, a smaller cosine similarity means a larger deviation (i.e., ϕ_k is quite unlike ϕ_{norm}), so ϕ_k should have a higher topic specificity (less general). In contrast, a larger cosine similarity means a smaller deviation (ϕ_k is more like ϕ_{norm}), so ϕ_k has a lower topic specificity (more general). With this idea, Topic Specification is computed by

$$TS(\phi_k) = 1 - \frac{\phi_k \cdot \phi_{norm}}{|\phi_k||\phi_{norm}|} \tag{9}$$

It can be seen that in Eq. (9), TS value is indeed inversely proportional to cosine similarity. A small TS value indicates a general topic, and a large TS value indicates a specific topic, which is also consistent with the literal meaning of Topic Specialization. In a topic hierarchy with a reasonable structure, a general topic near the root should have a small TS value, and in contrast, a specific topic towards the leaves should have a large TS value. In the experiment, different models are compared by the average TS at each level over all target domains with the total hierarchy level being 4. The experimental results are shown in Fig. 10 (Computer Science dataset) and Fig. 11 (Natural Science dataset).

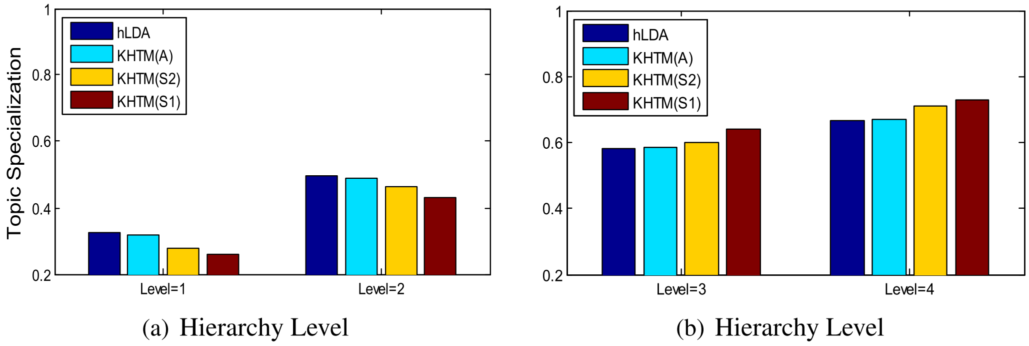
As demonstrated in Figs. 10 and 11, the proposed model KHTM (both in test settings 1 and 2) generates a more reasonable structure. In detail, in Fig. 10(a) and 11(a), it can be seen that the TS values of KHTM model are smaller, which means that the topics at level 1 and level 2 are more general. In Fig. 10(b) and 11(b), the TS values of KHTM model are larger, which means that the topics at level 3 and level 4 are more specific. Compared to the structure learned by KHTM, the topics of KHTM(A) at levels 1 and 2 are not general enough, and the topics at levels 3 and 4 are not specific enough. Empirically, without level constraint, the mined k-sets can indeed introduce noise. For example, as a general word, *learning* can form a k-set with word *classification* at level 2, while *classification* can also form a k-set with a specific word *kernel* at level 4. With no level constraint, *learning* will be promoted by *classification* at both levels, which lowers the specialization of the specific topic that contains *classification* at level 4.

5.6. Case study

In this section, we show the topic hierarchies learned by KHTM model in the two datasets. Fig. 12 shows the topic hierarchy that is learned from the abstracts corpus of conference *NIPS* in Computer Science dataset. Fig. 13 shows the topic hierarchy that is learned from the abstracts corpus of journal *Chemistry Central Journal* in Natural Science dataset. We show the top sub-topics for a supertopic. Each topic is represented by its top topical words. The total hierarchy level is 3. It can be seen that the structures of the two topic hierarchies are readily identified, and each topic is easily interpretable. Let us take the topic hierarchy of NIPS as an example (Fig. 12). The root topic contains the most general words, such as *analysis* and *framework*. The topics become more specific along with level increasing. For example, the topic *classification* (level 2) is followed by two sub-topics (level 3), i.e., *linear regression* and *learning to rank*, which are more specific.

5.7. Comparison of time and memory performance

To give a comprehensive view of the performance of the proposed KHTM model, in this section, we further report the comparison results of KHTM model and hLDA, first in running time and then in memory consumption. Let us first go back to KHTM model



(a) Hierarchy Level

(b) Hierarchy Level

Fig. 11. Topic Specialization Comparison (Natural Science dataset).

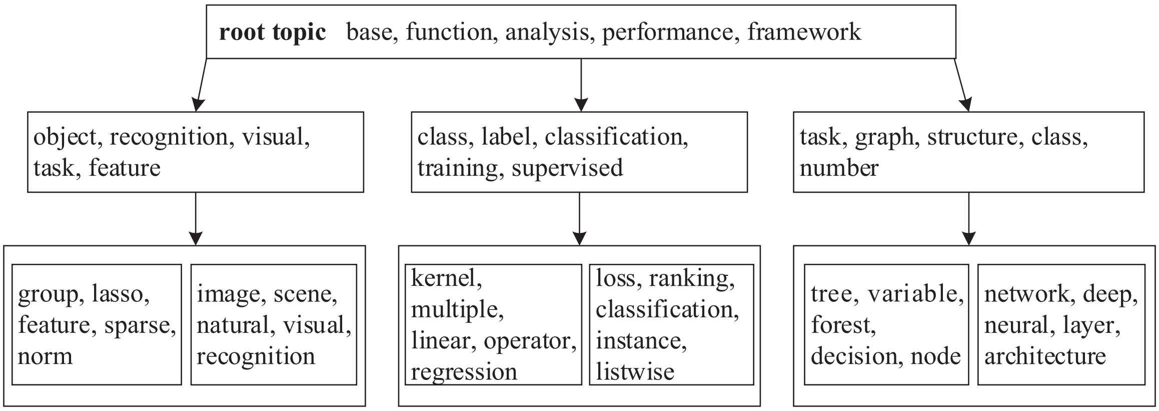

root topic base, function, analysis, performance, framework
object, recognition, visual, task, feature
class, label, classification, training, supervised
task, graph, structure, class, number
group, lasso, feature, sparse, norm
image, scene, natural, visual, recognition
kernel, multiple, linear, operator, regression
loss, ranking, classification, instance, listwise
tree, variable, forest, decision, node
network, deep, neural, layer, architecture

Fig. 12. The topic hierarchy learned from the abstracts corpus of NIPS.

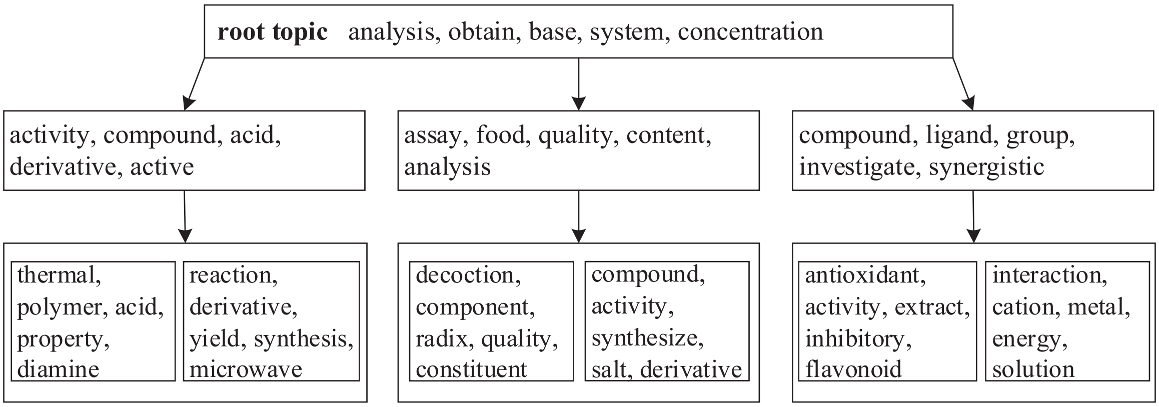

root topic analysis, obtain, base, system, concentration
activity, compound, acid, derivative, active
assay, food, quality, content, analysis
compound, ligand, group, investigate, synergistic
thermal, polymer, acid, property, diamine
reaction, derivative, yield, synthesis, microwave
decoction, component, radix, quality, constituent
compound, activity, synthesize, salt, derivative
antioxidant, activity, extract, inhibitory, flavonoid
interaction, cation, metal, energy, solution

Fig. 13. The topic hierarchy learned from the abstracts corpus of Chemistry Central Journal.

(Algorithm 2, Section 4). The running time of KHTM mainly comes from two sources. One is frequent itemset mining, and the other is Gibbs sampling. By contrast, hLDA does not have the capability of mining or using knowledge, so hLDA does not have the time consumption of frequent itemset mining. In Gibbs sampling, compared to hLDA, KHTM model does not introduce other new variables, so the extra running time comes from the utilization of promotion quantities, i.e., $\lambda_{l,w',w}$. We present the average running time of KHTM and hLDA over each domain corpus in Computer Science dataset and Natural Science dataset. The running time results are shown in Table 5, and the time unit is *second*.

The computer that is used to finish the experiments (also including the comparison of memory consumption) is with the following configuration: 1) CPU: Intel(R)Core(TM) i7-6500U@2.50Hz and 2.60Hz; 2) Memory: 8.0GB; 3) Operating System: Windows 10. The total hierarchy level is 3, and all parameters are set as default values.

Table 5
The average running time over each domain corpus (unit: second).

	KHTM	hLDA
Computer Science dataset	715.14	623.16
Natural Science dataset	617.87	532.31

Table 5 shows that KHTM consumes more time than hLDA, which is a natural result. But we also notice that, the extra time is limited, and KHTM consistently achieves higher topic quality and better topic specialization. Next we discuss the memory consumption. Compared to hLDA, the extra memory consumption mainly comes from two sources. One is the mined knowledge hierarchy (an example is given in Fig. 3) and the other is promotion quantities ($\lambda_{l,w',w}$). We report the average memory consumption of knowledge hierarchy and promotion quantities over each domain

Table 6
The average memory consumption over each domain corpus (unit: KB).

Dataset	knowledge hierarchy	promotion quantities	average corpus size
Computer Science	32.7	20.9	4126.7
Natural Science	16.6	12.0	3563.5

corpus. We also present the average size of each domain corpus, to enhance the comparative analysis. The experimental results are given in Table 6, and the unit is *KB (kilobyte)*.

From Table 6, it can be seen that, compared to the average corpus size, the memory consumption of knowledge hierarchy and promotion quantities is limited. Meanwhile, the memory consumption of promotion quantities is smaller than that of knowledge hierarchy. The reason is that we propose to use level constraint and PMI (pointwise mutual information) to filter those k-sets that are not suitable to be used in the target corpus. That is, not all k-sets in the mined knowledge hierarchy will be used.

6. Conclusion

In this paper, we proposed a novel hierarchical topic model KHTM that can mine knowledge from topic hierarchies of multiple domains corpora, and leverage such knowledge to learn a better topic hierarchy for the target domain. Also, we proposed an iterative algorithm that can improve the topic hierarchies in a continuous manner. The mined knowledge is organized in a hierarchical structure, and can be maintained and improved. We crawled two new multi-domain datasets. The evaluation results showed that the proposed KHTM model produced superior topic hierarchies, with more coherent topics and more reasonable hierarchical structures.

Acknowledgments

This paper is granted by Fundamental Research Fund for Central Universities (No. JBX171007), National Natural Science Fund of China (No.61702391), Zhejiang Provincial Natural Science Foundation (No.LY12F02003) and China Postdoctoral Science Foundation (No.2013M540492).

References

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In *Proceedings of international conference on very large data bases (VLDB)* (pp. 487–499).

Andrzejewski, D., Zhu, X., Craven, M., & Recht, B. (2011). A framework for incorporating general domain knowledge into latent dirichlet allocation using first-order logic. In *Proceedings of international joint conference on artificial intelligence (IJCAI)* (pp. 1171–1177).

Blei, D. M., Griffiths, T. L., & Jordan, M. I. (2010). The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM), 57*(2), 7:1–7:30.

Blei, D. M., Griffiths, T. L., Jordan, M. I., & Tenenbaum, J. B. (2005). Hierarchical topic models and the nested chinese restaurant process. *Advances in neural information processing systems (NIPS).*

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research (JMLR), 3*, 993–1022.

Chang, J., Boyd-Graber, J. L., Gerrish, S., Wang, C., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems (NIPS)* (pp. 288–296).

Chen, Z., Mukherjee, A., & Liu, B. (2014). Aspect extraction with automated prior knowledge learning. In *Proceedings of annual meeting of the association for computational linguistics (ACL)* (pp. 347–358).

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. In *Proceedings of the national academy of sciences of the united states of america (PNAS).*

Heinrich, G. (2008). Parameter estimation for text analysis. *Technical note.*

Jagarlamudi, J., DauméIII, H., & Udupa, R. (2012). Incorporating lexical priors into topic models. In *Proceedings of the conference of the European chapter of the association for computational linguistics (EACL 12)* (pp. 204–213).

Kang, J.-H., Ma, J., & Liu, Y. (2012). Transfer topic modeling with ease and scalability. In *Proceedings of SIAM international conference on data mining (SDM)* (pp. 564–575).

Kim, J. H., Kim, D., Kim, S., & Oh, A. (2012). Modeling topic hierarchies with the recursive chinese restaurant process. In *Proceedings of ACM conference on information and knowledge management (CIKM)* (pp. 783–792).

Kim, S., Zhang, J., Chen, Z., Oh, A., & Liu, S. (2013). A hierarchical aspect-sentiment model for online reviews. In *Proceedings of AAAI conference on artificial intelligence (AAAI).*

Mao, X.-L., He, J., Yan, H., & Li, X. (2012a). Hierarchical topic integration through semi-supervised hierarchical topic modeling. In *Proceedings of ACM conference on information and knowledge management (CIKM)* (pp. 1612–1616).

Mao, X.-L., Ming, Z.-Y., Chua, T.-S., Li, S., Yan, H., & Li, X. (2012b). Sshlda: a semi-supervised hierarchical topic model. In *Proceedings of international conference on empirical methods in natural language processing (EMNLP)* (pp. 800–809).

Mimno, D., Li, W., & McCallum, A. (2007). Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of international conference on machine learning (ICML)* (pp. 633–640).

Mimno, D. M., Wallach, H. M., Talley, E. M., Leenders, M., & McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of international conference on empirical methods in natural language processing (EMNLP)* (pp. 262–272).

Ming, Z.-Y., Wang, K., & Chua, T.-S. (2010). Prototype hierarchy based clustering for the categorization and navigation of web collections. In *Proceedings of ACM SIGIR* (pp. 2–9).

Mukherjee, A., & Liu, B. (2012). Aspect extraction through semi-supervised modeling. In *Proceedings of annual meeting of the association for computational linguistics (ACL)* (pp. 339–348).

Newman, D., Lau, J. H., Grieser, K., & Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Proceedings of annual conference of the North American chapter of the association for computational linguistics (NAACL)* (pp. 100–108).

Paisley, J., Wang, C., Blei, D. M., & Jordan, M. I. (2015). Nested hierarchical dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 37*(2), 256–270.

Pavlinek, M., & Podgorelec, V. (2017). Text classification method based on self-training and lda topic models. *Expert Systems with Applications (ESWA), 80*, 83–93.

Petinot, Y., McKeown, K., & Thadani, K. (2011). A hierarchical model of web summaries. In *Proceedings of annual meeting of the association for computational linguistics (ACL)* (pp. 670–675).

Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association, 101*, 1566–1581.

Wang, C., Danilevsky, M., Desai, N., Zhang, Y., Nguyen, P., Taula, T., & Han, J. (2013). A phrase mining framework for recursive construction of a topical hierarchy. In *Proceedings of ACM SIGKDD conference on knowledge discovery and data mining (KDD)* (pp. 437–445).

Wang, C., Liu, J., Desai, N., Danilevsky, M., & Han, J. (2014). Constructing topical hierarchies in heterogeneous information networks. *Knowledge and Information Systems (KAIS)*, 1–30.

Wang, S., Chen, Z., & Liu, B. (2016). Mining aspect-specific opinion using a holistic lifelong topic model. In *Proceedings of international world wide web conference (WWW)* (pp. 167–176).

Wu, Z., Lei, L., Li, G., Huang, H., Zheng, C., Chen, E., & Xu, G. (2017). A topic modeling based approach to novel document automatic summarization. *Expert Systems with Applications (ESWA), 84*, 12–23.

Xie, P., Yang, D., & Xing, E. P. (2015). Incorporating word correlation knowledge into topic modeling. In *Proceedings of annual conference of the North American chapter of the association for computational linguistics (NAACL)* (pp. 725–734).

Yan, X., Guo, J., Lan, Y., & Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of international world wide web conference (WWW)* (pp. 1445–1456).

Zhang, H., & Zhong, G. (2016). Improving short text classification by learning vector representations of both words and hidden topics. *Knowledge-Based Systems, 102*, 76–86.