Discovering Coherent Topics Using General Knowledge

Zhiyuan Chen[†], Arjun Mukherjee[†], Bing Liu[†], Meichun Hsu[‡], Malu Castellanos[‡], Riddhiman Ghosh[‡]
[†]University of Illinois at Chicago, [‡]HP Labs
{czyuanacm, arjun4787}@gmail.com, liub@cs.uic.edu, {meichun.hsu, malu.castellanos, riddhiman.ghosh}@hp.com

ABSTRACT
Topic models have been widely used to discover latent topics in text documents. However, they may produce topics that are not interpretable for an application. Researchers have proposed to incorporate prior domain knowledge into topic models to help produce coherent topics. The knowledge used in existing models is typically domain dependent and assumed to be correct. However, one key weakness of this knowledge-based approach is that it requires the user to know the domain very well and to be able to provide knowledge suitable for the domain, which is not always the case because in most real-life applications, the user wants to find what they do not know. In this paper, we propose a framework to leverage the general knowledge in topic models. Such knowledge is domain independent. Specifically, we use one form of general knowledge, i.e., lexical semantic relations of words such as synonyms, antonyms and adjective attributes, to help produce more coherent topics. However, there is a major obstacle, i.e., a word can have multiple meanings/senses and each meaning often has a different set of synonyms and antonyms. Not every meaning is suitable or correct for a domain. Wrong knowledge can result in poor quality topics. To deal with wrong knowledge, we propose a new model, called GK-LDA, which is able to effectively exploit the knowledge of lexical relations in dictionaries. To the best of our knowledge, GK-LDA is the first such model that can incorporate the domain independent knowledge. Our experiments using online product reviews show that GK-LDA performs significantly better than existing state-of-the-art models.

Categories and Subject Descriptors
H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing.

Keywords
Topic Models; General Knowledge; Lexical Relations

1. INTRODUCTION
Statistical topic models, such as pLSA [15] and LDA [5], provide a powerful framework for extracting latent topics in text documents. However, researchers have found that these unsupervised models often produce topics that are not interpretable or meaningful [30]. One key reason is that the objective functions of these models do not always correlate well with human judgments [9].

CIKM'13, October 27–November 01 2013, San Francisco, CA, USA.

http://dx.doi.org/10.1145/2505515.2505519

In order to deal with this problem, several knowledge-based models [1, 2, 17, 18, 31, 35] have been proposed. These models incorporate expert domain knowledge to guide the models. For example, the DF-LDA model [1] takes domain knowledge in the form of must-links and cannot-links. A must-link states that two words should belong to the same topic, while a cannot-link states that two words should not be in the same topic. In [8, 18, 25, 31], several seeded models were also proposed to allow the user to provide some prior seed words in some topics. The prior knowledge used in these existing models is typically domain dependent. For different applications, different knowledge is needed. These models also assume that the prior knowledge is correct.

These existing knowledge-based models have a key weakness to be applied in real life data mining applications, i.e., they all assume that the user knows the domain very well and can provide knowledge suitable for the domain, which is not always the case because in many real life data mining applications, the user wants to discover something unknown. The knowledge provided by the user usually needs to be repeatedly tuned in order to fit the domain (i.e., domain dependent knowledge) and improve the model.

In this paper, we propose to take a different approach. We believe that there is a vast amount of lexical knowledge about words and their relationships available in online dictionaries or other resources that can be exploited in a model to generate more coherent topics. Such knowledge is domain independent and can be easily extracted automatically from online dictionaries to form a general knowledge base. This knowledge base can serve as an integral part of a topic model system as it does not change from domain to domain and can be applied to any domain without any user involvement. To the best of our knowledge, this approach has not been taken before.

In this work, we utilize a specific type of lexical knowledge, i.e., *lexical semantic relations*, which are relations about words. Such relations include synonymy, antonym, hyponymy, taxonomy, meronymy, troponymy, adjective-attribute, etc. In order not to be too ambitious, in this first study we only use synonym, antonym and adjective-attribute relations to demonstrate the benefits of these relations to topic models[1]. Here adjective-attribute relation means an adjective and its attribute nouns that the adjective describes. For example, the adjective *expensive* usually describes the attribute *price* of an object. We believe that these lexical relations are beneficial to topic models in the sense that words in such relations are likely to belong to the same topic.

However, there is a major challenge in using lexical relations, i.e., many such relations may not be appropriate for a particular application because a word can have multiple meanings/senses. Each meaning/sense can have a different synonym set, a different antonym set and a different adjective-attribute set. For a particular

[1] Note that some antonyms can be useful (e.g., *expensive* and *cheap* both describe the topic "price") while some synonyms can be harmful (e.g., *picture* and *painting* may not fit coherently for topic "image" in the domain of digital cameras). Our model can automatically deal with these cases (see Section 3.3 for details).

application, typically only one or two meanings are applicable while the other senses are inappropriate. To make matters worse, even in the same sense, some words in the synonym set (also called *synset*) may be incorrect for a particular domain. For example, the word *picture* has 10 senses as a noun in WordNet [29]. The synset for the first sense is {*picture*, *image*, *icon*, *ikon*}. In the domain of digital cameras, *picture* and *image* should belong to the same topic, but *icon* and *ikon* should not share the same topic with *picture* and *image*. In the second sense of *picture*, the synset is {*picture, painting*}. These two words are not coherently related for cameras. The situation also exists in the other two lexical relations. To deal with it, we need a model that is able to automatically identify and leverage the right relations for a particular domain.

Before going further, we first describe how the lexical relations are represented in this work. We represent them as sets, e.g., synonyms: {expensive, pricey}, antonyms: {expensive, cheap}, and adjective-attributes: {expensive, price}. In our system, synonyms and antonyms are extracted from WordNet [29]. Adjective-attribute relations are obtained from the system in [11], which identifies such relations from online dictionaries. To simplify the presentation, we call these sets *LR-sets* (for lexical relation sets). Each LR-set indicates one sense/meaning of the words inside it. Since the WordNet [29] and the system in [11] can provide consistent sense IDs of each word, the synonyms, antonyms and adjective-attributes of the same sense for each word are automatically merged to form an LR-set. For example, for the word *expensive* in the above example, an LR-set {expensive, pricey, cheap, price} is automatically generated indicating one sense of this word. We will see in the Section 4 that LR-sets help improve resulting topics dramatically without any user involvement.

In [10], we proposed the MDK-LDA model to exploit the knowledge from multiple domains. However, MDK-LDA did not explicitly deal with wrong knowledge, especially the unfavorable incorrect semantic relationships in a particular domain. The knowledge used in [10] is called s-set which has the same structure as LR-set in this work. The major difference between s-set and LR-set is that s-sets were manually validated to ensure the quality of knowledge for each domain while the general knowledge in forms of LR-sets is domain independent and automatically extracted from the sources in [11, 29]. Thus, LR-sets contain much more noise, demanding additional mechanisms to explicitly deal with it. This paper proposes a comprehensive framework to incorporate the general (domain independent) knowledge with a new model called *GK-LDA* (short for *General Knowledge based LDA*).

Before discussing the proposed framework, we first review the two models of MDK-LDA, i.e., MDK-LDA(b) and MDK-LDA, which are the base of the proposed framework. In MDK-LDA(b), a new latent variable s is added into LDA to enable the model to choose a right LR-set ([10] used s-set) for each word. Each document is an admixture of latent topics while each topic is a probability distribution over LR-sets. MDK-LDA(b) suffers from the issue called *adverse effect of knowledge* [10], which also exists in other existing knowledge-based models. According to this issue, the frequent words may suffer from the attenuation of their probabilities when putting together with infrequent words as a piece of knowledge, resulting in uninterpretable output topics. To solve it, MDK-LDA employs the *generalized Pólya urn* (GPU) *model* [27] to promote LR-set as a whole. In MDK-LDA, drawing word w will not only increase the probability of seeing w, but also increase the probability of seeing words sharing an LR-set with w.

However, MDK-LDA is insufficient in terms of general knowledge represented by LR-sets due to its indiscrimination on each word in the LR-set. As mentioned before, some senses of a word may not be appropriate in a particular domain, leading to completely or partially incorrect LR-sets. Before applying LR-sets in topic models, we want to estimate the correlation between the domain corpus and LR-sets to have some ideas of the quality of LR-sets. If the domain corpus can validate an LR-set, we then can have a higher confidence in the usefulness of this knowledge, and hence trust it more. Based on this idea, we propose a matrix called *word correlation matrix* which estimates the quality of an LR-set by validating the co-occurrences of the words in the LR-set in the corpus (represented by word probabilities under topics in LDA). In more details, the original LDA (without any knowledge) is executed on the corpus at first. The resulting word distributions under output topics of LDA are then used to estimate the correctness of the relationships in the LR-sets in order to reduce the undesirable effects of them. The intuition is that the words in an LR-set may be less likely to be in the same topic if they have very different probability masses (too far away from each other in the order of word probabilities) under LDA's output topics. With this estimation of LR-sets, we propose the GK-LDA model by employing a new GPU model with a new Gibbs sampler which can use the proposed word correlation matrix to discriminate the words in each LR-set. In GK-LDA, drawing word w will not only promote the LR-set as a whole, but also discriminately promote each of the correlated words according to the word correlation matrix.

Our experimental evaluation was conducted using online reviews from four domains. Our results show that GK-LDA outperforms several state-of-the-art baseline models by a large margin.

In summary, this paper makes the following contributions:

1. It proposes the idea of exploiting the general knowledge of lexical semantic relations in topic models to produce coherent topics automatically. To the best of our knowledge, this is the first paper that systematically studies such domain independent knowledge in topic models with the aim to provide a general platform to be used in any application domain. Although there are some existing models that use synonyms or other dictionary information in topic models, they are limited in scope and different from our work as we will see in Section 2.
2. It proposes a novel model called GK-LDA. We believe that GK-LDA is the first model that tries to explicitly deal with the problem of wrong input knowledge for an application domain.
3. A comprehensive evaluation has been conducted to compare the proposed framework with several state-of-the-art baselines based on various qualitative and quantitative measures. The results using the automatically generated knowledge demonstrate the superiority of GK-LDA.

2. RELATED WORK

Topic models, such as pLSA [15] and LDA [5], have been successfully applied to many applications, e.g., sentiment analysis [41] and analysis of discussion threads [24]. However, these methods do not use any prior knowledge or external resources.

In recent years, researchers have proposed to incorporate external resources to guide the modeling process. These resources include temporal information [16], social network [28], citation resources [20], search results [33], bilingual resources [42], etc. But these resources are mainly for some specific tasks and not widely available. Some other researchers also tried to leverage domain de-

pendent knowledge into topic models [1, 2, 8, 13, 17, 18, 25, 31, 35], which typically assume the knowledge is correct and specific to a particular domain. The key weakness of this knowledge-based approach has been discussed in Section 1. A representative model in this regard is DF-LDA [1], which allows the user to set must-links and cannot-links (see Section 1). DF-LDA also assumes that must-links and cannot-links are correct. Further, the definition of must-link is transitive, i.e., if A and B form a must-link, and B and C form a must-link, it implies a must-link between A and C, indicating A, B and C should be in the same topic. This definition can link unrelated words together due to multiple senses, giving poor results in the case of general knowledge. In our work, several LR-sets may share words but they do not have to be transitive. In [2], the authors extended the work to allow more general knowledge in the form of first-order logic. Along the same line, several seeded models [8, 18, 25, 31] have been proposed. A seed set, which is a set of seed words for a topic, can also be expressed as must-links. Like DF-LDA, they also assume the user knowledge is correct.

Some other related works include LDAWN [6], which incorporates WORDNET-WALK in modeling. Their synsets are similar to LR-sets. However, their task is word sense disambiguation. LDAWN assumes that a word is generated by a WordNet-Walk in WordNet [29], while in our case, a word is generated by its latent topic and LR-set, where each topic is a multinomial distribution over LR-sets. One interesting result they found is that idiosyncrasies in the hierarchical structure of WordNet can harm performance [23], which is consistent with our intuition that simply incorporating general lexical knowledge can degrade the performance of the model if we do not have a mechanism to deal with wrong knowledge. [35] employs the multi-language topic synchronization. The dictionary information is used to bias corresponding words towards similar topics. However, it also assumes that the knowledge is correct and domain specific. It again cannot handle wrong knowledge. [4, 36] use document label information in a supervised setting. Our framework does not use supervision. In [26], a semi-supervised model was proposed, which is very different from our work as they use expert reviews to guide the analysis of user reviews. The model in [17] also enables the user to provide knowledge interactively during the modeling process.

The *generalized Pólya urn* (GPU) model [27] was first utilized in LDA to generate coherent topics in [30]. However, [30] did not use any external resources or knowledge. In addition, its objective of applying GPU model is different in the sense that [30] uses it to smooth the probabilities of words with co-document frequencies while our objective is to promote the LR-sets and the correlated words. As we will see in Section 4, its results are inferior to GK-LDA model. The GPU model also does not deal with wrong knowledge. Our proposed GK-LDA model, to the best of our knowledge, is the first model to exploit general knowledge (which is domain independent) and deal with wrong knowledge explicitly.

3. THE PROPOSED TECHNIQUES

3.1 Lexical Semantic Relations

In this section, we detail lexical semantic relations (LSR), and their representations. LSRs are relations about words. There are many types of LSRs, e.g., synonymy, antonym, hyponymy, taxonomy, adjective-attribute, and others [3]. As we noted earlier, we only use synonymy, antonym and adjective-attribute relations in this paper to investigate the benefits of leveraging such knowledge in topic models.

Synonymy: Two expressions *a* and *b* of a language are synonyms *iff* they mean exactly or nearly the same. The notion is typically applied to lexical items, including idioms, but it can be used for larger expressions as well. In this work, we only use the word level synonyms, e.g., *expensive* and *pricey*.

Antonym: Two expressions *a* and *b* of a language are antonyms *iff* they have opposite meanings. Again, the notion can be words or larger expressions. In this work, we only use the word level antonyms, e.g., *expensive* and *cheap*.

Adjective-attribute: An adjective is a word that modifies nouns and pronouns, primarily by describing a particular quality/attribute of the word it is modifying. Although there are some general adjectives which can describe/modify anything, e.g., *good* and *bad*, most adjectives describe some specific attributes or properties of nouns. For example, *expensive* usually describes *price*, and *beautiful* often describes *appearance*.

As we described earlier, the LSRs are represented as LR-sets where each LR-set is automatically generated by merging synonyms, antonyms and adjective-attributes of the same word sense from the sources of [11, 29].

3.2 The MDK-LDA Model

3.2.1 MDK-LDA(b)

We first introduce the basic model MDK-LDA(b). A new latent variable s, which denotes the LR-set assignment to each word, is added into LDA. The generative process is given by:

1. For each topic $t \in \{1, \dots, T\}$
   i. Draw a per topic distribution over LR-sets, $\varphi_t \sim Dir(\beta)$
   ii. For each LR-set $s \in \{1, \dots, S\}$
      a) Draw a per topic, per LR-set distribution over words, $\eta_{t,s} \sim Dir(\gamma)$
2. For each document $m \in \{1, \dots, M\}$
   i. Draw a topic distribution per document, $\theta_m \sim Dir(\alpha)$
   ii. For each word $w_{m,n}$, where $n \in \{1, \dots, N_m\}$
      a) Draw a topic $z_{m,n} \sim Mult(\theta_m)$
      b) Draw an LR-set $s_{m,n} \sim Mult(\varphi_{z_{m,n}})$
      c) Emit word $w_{m,n} \sim Mult(\eta_{z_{m,n},s_{m,n}})$

where M is the number of documents and each document m has N_m words. The vocabulary in the corpus is denoted by $\{1, \dots, V\}$ and the number of LR-sets is S. We follow [10] to set the smoothing hyperparameter γ as the exponential function to control the density of Dirichlet distribution:

$$\gamma_s = \lambda \cdot e^{-|s|} \tag{1}$$

3.2.2 MDK-LDA

In order to avoid the attenuation of the probability masses of the frequent words in MDK-LDA(b), MDK-LDA employs the generalized Pólya urn (GPU) model [27, 30] to promote an LR-set as a whole. In Pólya urn models, objects of interest are represented as colored balls in an urn. In a simple Pólya urn model, when a ball of a particular color is drawn, that ball is replaced with two balls of the same color in the urn. In contrast, in the GPU model, when a ball is drawn, that ball is put back along with a certain number of balls of similar colors. More details can be found in [10]. In order to promote an LR-set upon observing any of its word, if a ball of color w is drawn, $\boldsymbol{A}_{s,w',w}$ balls of each color $w' \in \{1, \dots, V\}$ are put back where w and w' share LR-set s. $\boldsymbol{A}_{s,w',w}$ is defined as:

$$\boldsymbol{A}_{s,w',w} = \begin{cases} 1 & w = w' \\ \sigma & w \in s, w' \in s, w \neq w' \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

3.3 The GK-LDA Model

Since our LR-sets are general knowledge from online dictionaries, some LR-sets do not make sense in a particular domain. For example, {card, bill} is a correct LR-set in the domain "Restaurant", but unsuitable in the domain "Camera." In this section, we propose the GK-LDA model to deal with wrong LR-sets, which can be applied in any application domain. Note that GK-LDA still uses the graphical model of MDK-LDA.

3.3.1 Issues of Wrong LR-sets

MDK-LDA(b) and MDK-LDA are able to deal with some incorrect LR-sets owning to their ability to choose the LR-set with the right word sense in the modeling. However, there are two major issues that these two models are unable to deal with.

1. One issue is that there may be no correct LR-sets for a word in an application domain. That is, for a word w, all LR-sets containing w do not make sense (are wrong) for the domain. In this case, the model does not have any correct LR-sets to choose from. As a result, the assigned LR-sets to w will all be incorrect, leading to promotion of LR-sets that have words that are not semantically related in a particular domain.
2. The other issue is that an LR-set may be partially correct and partially incorrect in a domain, meaning that some words in the LR-set do not share a similar semantic meaning with others for a particular domain. For example, in the domain "Camera", we have an LR-set {picture, pic, flick} where *picture* shares a similar meaning with *pic*, but both of them have different semantic meaning from *flick*. In this case, when we promote this LR-set as in the model MDK-LDA, we may promote all the relationships inside it, including the wrong ones: picture-flick and pic-flick. As a result, it may lead to merging of words with different semantic meanings, which results in multiple sub-topics inside one topic.

To solve these two issues, we propose the GK-LDA model. First, we propose a word correlation matrix $\boldsymbol{C}$ to estimate the correctness of the LR-sets using the given corpus (Section 3.3.2). Using this matrix, for each word w, we relax the constraints of all wrong LR-sets (the first issue above) by adding a singleton LR-set $\{w\}$ (Section 3.3.3). To deal with the second issue, we scale the matrix $\boldsymbol{C}$ into a matrix $\boldsymbol{C}'$, which fits in the new GPU model in GK-LDA (Section 3.3.4). This new matrix $\boldsymbol{C}'$ is used to design a new Gibbs sampler for the GK-LDA model (Section 3.4).

3.3.2 Word Correlation Matrix

Given a piece of knowledge (LR-set) itself, we may not have any idea whether it is correct or not. However, given a corpus, it is possible to validate the LR-set through the corpus. If an LR-set has a reasonable support in the corpus, we will have some confidence in its usefulness in the domain represented by the corpus, and consequently we give it a higher weight for promotion. Since the topics found by LDA are a reasonable summary of the corpus and the top words (with high probabilities) under each topic are more likely to share some semantic similarity, we use the topic-word distribution from LDA to estimate word correlations in each LR-set in a domain. The idea is that if two words in an LR-set are too far from each other (i.e., have very different probabilities) under the topics of LDA, they are more likely to have different semantic meanings, i.e., less correlated.

Figure 1 gives a detailed algorithm for computing word correlation matrix $\boldsymbol{C}$. Intuitively, top words, with higher probabilities under a topic, are more likely to represent the semantic concept of the topic while words with low probabilities contribute much less

Algorithm Computing Word Correlation Matrix $\boldsymbol{C}$
Input: Word distribution under topics φ_t from LDA
LR-sets $\{1, \dots, S\}$
Output: $\boldsymbol{C}$
1 **For** each LR-set $s \in \{1, \dots, S\}$
2 **For** each pair $(w, w') \in s$
3 $P_{max}(w) = max_{t \in \{1 \dots T\}} \varphi_t(w)$;
4 $P_{max}(w') = max_{t \in \{1 \dots T\}} \varphi_t(w')$;
5 **If** $P_{max}(w) > P_{max}(w')$ **then**
6 **Exchange** w and w';
7 $t_{max} = argmax_{t \in \{1 \dots T\}} \varphi_t(w')$;
8 $\boldsymbol{C}_{s,w',w} = \frac{\varphi_{t_{max}}(w)}{\varphi_{t_{max}}(w')}$;
9 **Return** $\boldsymbol{C}$;

Figure 1. Computing Word Correlation Matrix $\boldsymbol{C}$.

to the semantic concept. To compute the correlation of two words, we focus on the topics where the words have high probabilities. The algorithm in Figure 1 computes for all the word pairs (w, w') in each LR-set (lines 1 and 2). The word distribution under topic t is denoted by φ_t in LDA. For those pairs not in any LR-set, their correlation is 0 (not shown in the algorithm) and we do not need to validate them. Lines 3 and 4 find the topics that the two words w and w' have the maximum probabilities respectively. Lines 5 and 6 enforce that word w has a lower (or equal) maximum probability than w', restricting their ratio not larger than 1. Line 7 finds the topic that word w' has the maximum probability, and the ratio of probabilities of both words under this topic is estimated as the correlation (Line 8). The idea is that the ratio of word probabilities under this topic is a good indicator of the semantic correlation of the two words. Although this word correlation estimation may not be perfect due to the imperfect topic-word distributions from LDA, our experiments show that it is effective in solving the two issues discussed in Section 3.3.1.

3.3.3 Relaxing Wrong LR-sets

In order to solve issue 1 mentioned in Section 3.3.1, we need to design a function to estimate the quality of LR-set s toward the word w. Since the quality of s depends on the correlations of words inside it with w, we can estimate the quality of LR-set s towards w based on the word correlation matrix $\boldsymbol{C}$ as follows:

$$Q(s, w) = \begin{cases} \max\limits_{w' \in s, w \neq w'} \boldsymbol{C}_{s,w',w} & w \in s \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Basically, the quality function of LR-set s towards word w is the maximum correlation between any word w' ($w' \in s$ and $w' \neq w$) and w based on $\boldsymbol{C}$. This quality function can give us some hints as to which LR-sets are more likely to be correct or incorrect. We set a threshold ε (discussed in Section 4) such that if the quality of an LR-set towards each word inside it is less than ε, this LR-set is estimated to be wrong (or low-quality) in the domain. Following the issue 1, if all LR-sets of word w are estimated to be wrong, we need to add an alternative LR-set to give the model a right LR-set to choose. In this case, a singleton LR-set (i.e., $\{w\}$) is added to relax the LR-set constraint. If w has any LR-set with its quality value greater than ε, the singleton set $\{w\}$ is not added. This preprocessing ensures that the model can have at least one reasonable LR-set to assign to each word. However, note that, the estimated wrong LR-sets are *not removed* because the estimation above on topics generated by LDA may not be perfect.

3.3.4 Incorporating Correlation Matrix

We now deal with issue 2 discussed in Section 3.3.1, i.e., the partial incorrect LR-sets. In MDK-LDA, when a word is drawn, all

other words inside the LR-set will be put back equally according to matrix $\boldsymbol{A}$ in equation 2, which promotes the LR-set as a whole, i.e., promoting every word inside it. Now we want to use the word correlation values of $\boldsymbol{C}$ to help determine the number of balls to put back which reduces the undesirable effects of wrong relationships in an LR-set. For this purpose, we scale $\boldsymbol{C}$ to $\boldsymbol{C'}$ as follows, which will be incorporated in the GK-LDA model.

$$\boldsymbol{C'}_{s,w',w} = \begin{cases} 1 & w = w' \\ \tau \cdot \boldsymbol{C}_{s,w',w} & w \in s, w' \in s, w \neq w' \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The coefficient τ governs the scale of correlation corresponding to the hyperparameters β and γ in the model. The value of τ will be discussed in Section 4. With the matrix $\boldsymbol{C'}$, we can design a new GPU model, i.e., drawing word w will not only increase the probability of seeing w, but also discriminatively increase the probability of seeing every correlated word with w represented by $\boldsymbol{C'}$. Following the example of LR-set s = {picture, pic, flick} in Section 3.3.1, since *picture* and *pic* are semantically related in the domain "Camera" (in other words, the relationship picture-pic is correct), they tend to have reasonable high co-occurrence in the corpus and hence LDA is likely to put them together under the same topic. On the other hand, *flick* is semantically different from both *picture* and *pic*, and thus LDA may put *flick* under a different topic. As a result, $\boldsymbol{C'}_{s,"picture","pic"}$ will be much larger than $\boldsymbol{C'}_{s,"picture","flick"}$ and $\boldsymbol{C'}_{s,"pic","flick"}$. In the GPU model of GK-LDA, seeing the word *picture* and *pic* will promote each other a lot, but promote the word *flick* very little, which is consistent with our aim to merge the semantically related words while separating semantically different words.

3.4 Inference

In this section, we introduce a new Gibbs sampler for the GK-LDA model. The Gibbs samplers for MDK-LDA(b) and MDK-LDA can be found in [10].

In topic models, collapsed Gibbs sampling [12], one of Markov Chain Monte Carlo (MCMC) methods [37], is a standard procedure for obtaining a Markov chain over the latent variables in the model. In GK-LDA, the latent variables (i.e., latent topic z and latent LR-set s) are jointly sampled, which gives us a blocked Gibbs sampler. An alternative way is to perform hierarchical sampling (sample z and then s). However, [38] argues that when the latent variables are highly related, blocked samplers improve convergence of the Markov chain and also reduce autocorrelation.

As in Section 3.3, the GK-LDA model employs the GPU model to promote each correlated word represented by the matrix $\boldsymbol{C'}$. However, the GPU model is nonexchangeable, meaning that the joint probability of the words in any given topic is not invariant to the permutation of those words. Inference of $\boldsymbol{z}$ and $\boldsymbol{s}$ can be computationally expensive due to the non-exchangeability of words. We take the approach of [30] which approximates the true Gibbs sampling distribution by treating each word as if it were the last. When sampling a word w, we first promote the LR-set of it as a whole. Then, each word in the LR-set is promoted based on its word correlation with w according to $\boldsymbol{C'}$. The idea is that if a word w' is more correlated with w, it should be promoted more when w is seen, pushing them into the same topic. Note that promotion in the GPU model is achieved by putting back balls of the corresponding colors into the urn. Denoting the random variable $\{z, s, w\}$ by singular subscripts $\{z_i, s_i, w_i\}$, where i denotes the variable corresponding to each word in each document in the corpus, the conditional probability to assign a topic t and an LR-set s (containing the word w_i) to the word w_i is given by:

$$P\left(z_i = t, s_i = s \middle| \boldsymbol{z}^{-i}, \boldsymbol{s}^{-i}, \boldsymbol{w}, \alpha, \beta, \gamma, \boldsymbol{C'}\right) \propto \frac{n_{m,t}^{-i} + \alpha}{\sum_{t'=1}^{T}\left(n_{m,t'}^{-i} + \alpha\right)} \times \frac{\sum_{w'=1}^{V}\sum_{v'=1}^{V} \boldsymbol{C'}_{s,v',w'} \cdot n_{t,s,v'}^{-i} + \beta}{\sum_{s'=1}^{S}\left(\sum_{w'=1}^{V}\sum_{v'=1}^{V} \boldsymbol{C'}_{s',v',w'} \cdot n_{t,s',v'}^{-i} + \beta\right)} \times \frac{\sum_{w'=1}^{V} \boldsymbol{C'}_{s,w',w_i} \cdot n_{t,s,w'}^{-i} + \gamma_s}{\sum_{v'=1}^{V}\left(\sum_{w'=1}^{V} \boldsymbol{C'}_{s,w',v'} \cdot n_{t,s,w'}^{-i} + \gamma_s\right)} \quad (5)$$

where n^{-i} denotes the count excluding current assignment of z_i and s_i, i.e., $\boldsymbol{z}^{-i}$ and $\boldsymbol{s}^{-i}$. $n_{m,t}$ denotes the number of occurrences that topic t was assigned to word tokens in document m. $n_{t,s}$ denotes the count that LR-set s occurs under topic t. $n_{t,s,v}$ refers to the number of times that word v appears in LR-set s under topic t.

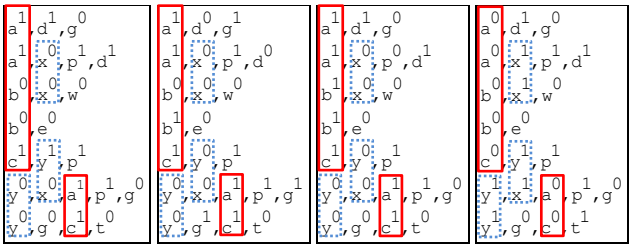
3.5 Model Comparison Using an Example

In order to illustrate the differences between LDA, MDK-LDA(b), MDK-LDA, and GK-LDA, we create an artificial corpus (Figure 2) and two LR-sets: {a, b, c} and {x, y}. Each letter in the corpus represents a word token. The hyperparameters α and β are set to the value suggested in [12]. For other parameters, we empirically set $\lambda = 2000$, $\sigma = 0.2$ and $\tau = 1$ for the purpose of demonstrating the mechanisms of our framework. We run collapsed Gibbs sampling 1000 iterations with 2 topics in total.

Sample from Markov chain: In Figure 2, we show the final topic assignment to each word (in superscript) based on the final Markov chain status for LDA (Figure 2(1)), MDK-LDA(b) (Figure 2(2)), MDK-LDA (Figure 2(3)), and GK-LDA (Figure 2(4)). It is clear that since LDA cannot use any external knowledge, it splits a, b and c into different topics. Additionally, it assigns topic 1 to y in line 5, which is different from other y's (and x's). On the other hand, our framework can leverage the information from LR-sets. MDK-LDA(b) drags b in line 4 to the same topic as a and c (i.e., topic 1), as well as corrects the topic assignment of y in line 5. Both MDK-LDA and GK-LDA successfully groups all of {a, b, c} into one topic and all of {x, y} into the other topic (in red solid and blue dashed rectangles).

Topic-word distribution: Table 1 shows the top 7 words under each topic ranked by their probabilities (round to 3 decimal places). The following interesting observations explain the effectiveness of our framework:

1. Since LDA cannot exploit knowledge, b has high probability under topic 0 while a and c have high probability under topic 1, which is undesirable as they are related according to the knowledge. In addition, y has lower probability than x under topic 0 since one y appears under topic 1 as mentioned above.
2. For MDK-LDA(b), we find that a, b and c are close to each other in the word ranking, as well as x and y. This is exactly what MDK-LDA(b) aims to achieve, i.e., make the probability of words in the LR-sets closer to each other. However, it has the issue *adverse effect of knowledge* [10]. In Table 1, we can see that under topic 0, a with $prob = 0.229$ is ranked lower than other words also appeared 3 times in the corpus (e.g., g and p with $prob = 0.242$). Since a, b and c share similar semantic meaning, we can semantically treat them as one word (say a_b_c) under topic 0. Then, the occurrence of a_b_c is 6 under topic 0, which is higher than any other word with 3 times (e.g. g and p) under topic 0. Following this intuition, as individual word, a should rank slightly higher than g and p rather than lower. In real data sets (Section 4), this issue also causes the drop of precision of top words (see Section 4.3).



(1) (2) (3) (4)

Figure 2. Model comparison using an example

(1): LDA. (2): MDK-LDA(b). (3) MDK-LDA. (4) GK-LDA.

The corpus is (1) without superscripts. Each superscript is the topic ID that is assigned to each word by each model.

Table 1. Words ranked by probability under the topic.

LDA				MDK-LDA(b)			
Topic 0		Topic 1		Topic 0		Topic 1	
W	P	W	P	W	P	W	P
g	0.220	a	0.256	x	0.238	g	0.242
x	0.220	p	0.256	y	0.238	p	0.242
b	0.150	d	0.174	d	0.164	a	0.229
y	0.150	c	0.174	w	0.086	c	0.159
w	0.078	y	0.091	e	0.086	b	0.089
e	0.078	g	0.008	t	0.086	d	0.008
t	0.078	x	0.008	b	0.061	w	0.008

MDK-LDA				GK-LDA			
Topic 0		Topic 1		Topic 0		Topic 1	
W	P	W	P	W	P	W	P
x	0.243	a	0.284	a	0.266	x	0.301
y	0.243	b	0.197	c	0.250	y	0.301
g	0.207	c	0.197	g	0.187	p	0.184
p	0.073	d	0.144	b	0.139	d	0.125
w	0.073	p	0.144	w	0.067	t	0.065
e	0.073	g	0.007	e	0.067	g	0.006
t	0.073	w	0.007	d	0.006	w	0.006

3. MDK-LDA solves the above problem by promoting the LR-sets. We can see that, in Table 1, x and y rank higher than g that also has three occurrences under topic 0. This is more intuitive as x and y share similar semantic meaning and both show up under topic 0, which should increase the probability of seeing either of x or y under this topic. The same problem for a, b and c is also solved.
4. MDK-LDA(b) and MDK-LDA follows the knowledge assuming that it is correct. However, the knowledge is not always equally useful in a particular domain. As the example above is a toy corpus, we cannot strongly claim the helpfulness of the knowledge. Nonetheless, we can make a reasonable assumption that the relationship a-c should be stronger than a-b and b-c since LDA puts a and c under one topic originally. In other words, the correlation in a-c is more consistent with the corpus than a-b and b-c. Consequently, although b and c both appear twice under topic 0, we assign higher probability to c ($prob = 0.250$) and lower probability to b ($prob = 0.139$), which reflects a balance between the corpus and the knowledge.

The example in this section shows that our framework can effectively use the knowledge. However, it is hard to show wrong knowledge with such a small corpus. Our experiments with real-life datasets in the next section will show the benefits of GK-LDA, which improves MDK-LDA.

Table 2. Corpus statistics and LR-sets statistics.

Domain	Reviews	Sentences	Words	LR-sets	Estimated Wrong LR-sets by GK-LDA
Camera	500	5171	947	432	113
Cellphone	500	2527	385	174	35
Food	500	2416	459	204	41
Computer	500	2864	496	223	49
Average	500	3245	572	258	60

4. EXPERIMENTS

We now evaluate the proposed GK-LDA model, as well as MDK-LDA(b) and MDK-LDA, and compare them with three baseline models: LDA [5], LDA with GPU (denoted as LDA-GPU) [30] and DF-LDA [1]. LDA is the basic knowledge-free unsupervised topic model. LDA-GPU applied GPU in LDA using co-document frequency. DF-LDA is perhaps the most well-known knowledge-based model which introduced must-links and cannot-links. It is also a natural fit for our proposed model as a must-link and an LR-set share the similar notion, i.e., they both aim at constraining the words in them to appear under the same topic. Note that existing models typically assume that the knowledge is correct and to our knowledge there is no prior work in topic modeling that can deal with wrong knowledge explicitly. Our proposed GK-LDA model can deal with wrong knowledge. We will see in Sections 4.2 and 4.3 that this capability of GK-LDA results in far better results than existing state-of-the-art models.

In Section 4.1, we describe the datasets and experimental settings. In Section 4.2, we evaluate our framework objectively using the Topic Coherence metric [30] and KL-Divergence. Further, in Section 4.3.1, we report the human evaluation results by working with two judges who are familiar with the Amazon products and reviews. Last, we show the qualitative results with some example topics from different models in Section 4.3.2.

4.1 Datasets and Settings

Datasets: Since LR-sets and the proposed framework are domain independent mechanisms for finding topics from text collections, we use multiple datasets from different domains of online reviews for our evaluation. We collected reviews from four domains from Amazon.com. Each domain collection (or corpus) contains 500 reviews. The statistics of each domain are shown in Table 2 (columns 2, 3, and 4). The four domains are "Camera & Photo", "Cell Phones & Accessories", "Gourmet Food & Grocery", and "Computers & Accessories". For easy presentation, we simply use "Camera", "Cell Phone", "Food", and "Computer" to denote the four domain corpora respectively. We have made the datasets publically available at the website of the first author.

Pre-processing: We ran the Stanford Parser[2] to perform sentence detection, lemmatization, and POS tagging. Then, punctuations, stop words[3], numbers and words appearing less than 5 times in each corpus were removed. For each domain, the domain name was also removed as it appears very frequently and co-occurs with most words in the corpus, leading to high similarity among topics. Our LR-sets depend on POS tags of words. In this work, we only use nouns and adjectives to produce LR-sets since they are the main parts of the topics. Verbs have a high level of noise. We plan to consider verbs based LR-sets in our future work. The number of LR-sets (having at least two words) and the number of estimated wrong LR-sets (having at least two words) by GK-LDA (in

[2] http://www-nlp.stanford.edu/software/corenlp.shtml

[3] http://jmlr.csail.mit.edu/papers/volume5/lewis04a/a11-smart-stop-list/english.stop

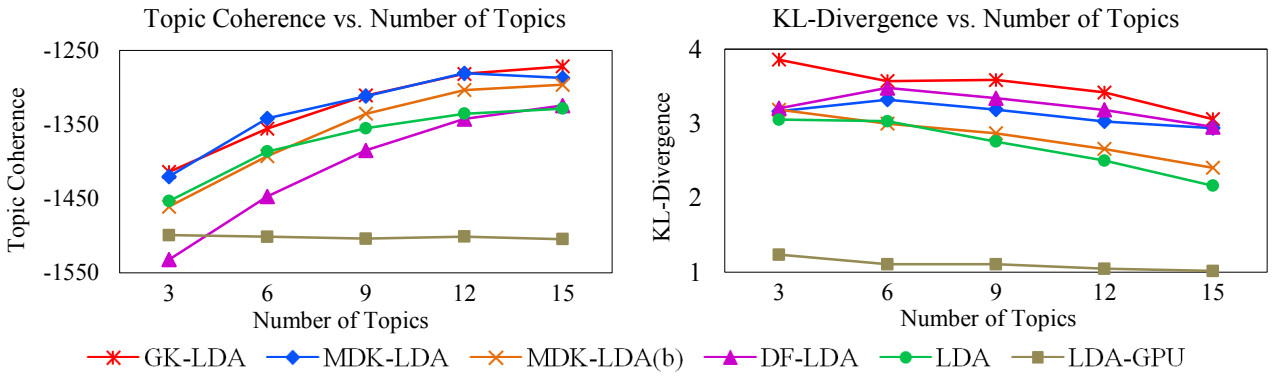


Figure 3. (Left) Average Topic Coherence score of each model, and (Right) Average KL-Divergence of each model.

Section 3.3.3) in each domain are given in Table 2 (columns 5 and 6). Note that duplicate LR-sets have been removed.

Sentences as documents: As pointed out in [41], when standard topic models are applied to reviews, they tend to produce topics that correspond to global properties of product, which make topics overlapping with each other. Since applying topic models to reviews mainly aims to find different aspects or features (as topics) of products [19, 31, 41, 43], using individual reviews for modeling is not very effective [41]. Although there are models dealing with sentences in complex ways [19, 41], we take the approach in [7], dividing each review into sentences and treating each sentence as a document. Sentences can be used by all three baselines without any change to their models. Although the relationship between sentences of a review is lost, the data is fair to all models.

Parameter settings: For all models, posterior inference was drawn after 1000 Gibbs iterations with an initial burn-in of 100 iterations. For all models, we set $\alpha = 1$ and $\beta = 0.1$. We found that small changes of α and β did not affect the results much, which was also reported in [19] who also used online reviews. For the number of topics, we tried different values (see Section 4.2.1). Note that it is difficult to know the exact number of topics. While non-parametric Bayesian approaches [40] aim to estimate the number of topics from the corpus, they are often sensitive to hyperparameters [14]. In this work, the heuristic values obtained from our experiments produced good results.

For DF-LDA, we followed the definition of must-link to generate must-links from LR-sets. LR-sets don't contain cannot-link knowledge. Note that the generated must-links contain wrong knowledge due to the issue of multiple senses, which degrades the performance of DF-LDA as we will see in Sections 4.2 and 4.3. We then ran DF-LDA (implementation downloaded from its authors' website) while keeping the parameters as proposed in [1] (we also experimented with different parameter settings but they did not produce better results). For our framework, we empirically set $\lambda = 2000$. For the threshold of ε, in GK-LDA, we estimated it using some labeled LR-sets in a development corpus, "Watch", which was not used in the evaluation (different from the four domains in Table 2). Based on the labeled data, we empirically chose the threshold $\varepsilon = 0.07$, meaning that if the quality of LR-set is lower than this value, we will add a singleton set as described in Section 3.3.3. We then averaged the word correlation values (Figure 1) of word pairs in the estimated correct LR-sets to set $\sigma = 0.2$ and $\tau = 2$ in equations 2 and 4. Although these three parameters come from the domain "Watch", we use them for all four domains in Table 2. Another approach is to automatically determine the threshold ε to distinguish correct and incorrect LR-sets. We defer this to our future work.

4.2 Objective Evaluation

In this section, we evaluate our framework objectively. Topic models are often evaluated using perplexity on held-out test data. However, the perplexity measure does not reflect the semantic coherence of individual topics learned by a topic model [34]. Recent research has shown potential issues with perplexity as a measure: [9] suggested that human judgments can sometimes be contrary to the perplexity measure. Also, perplexity does not really reflect our goal of finding coherent topics with accurate semantic clustering. It only provides a measure of how well the model fits the data. Thus, we choose two evaluation metrics, Topic Coherence and KL-Divergence, which directly evaluate our framework on topic interpretability and topic distinctiveness [21, 32]. We also report statistical significance of improvements of our framework calculated based on paired *t*-test.

Topic Coherence: The *Topic Coherence* metric [30] (also called UMass measure [39]) was proposed for assessing topic quality. The metric relies upon word co-occurrence statistics within the documents, and does not depend on external resources or human labeling. [30] shows that topic coherence is highly consistent with human labeling. Higher Topic Coherence score indicates higher quality of topic, i.e. better topic interpretability.

KL-Divergence: Another important metric for topic models is topic distinctiveness [21, 32]. We want to evaluate how distinctive the discovered topics are. To measure the distinctiveness, we use *KL-Divergence* as in [21, 32]. Since KL-Divergence is asymmetric, we compute its values between all pairs of topics and average them to get the average KL-Divergence. Clearly, for more distinctive topic discovery and better topic quality, it is desirable to have larger average KL-Divergence.

4.2.1 Effects of Number of Topics

Since the models in our experiments are all parametric topic models, we first compare the performance of each model given different number of topics. Figure 3 shows the average Topic Coherence score and KL-Divergence (over all domains) of each model given different number of topics. We can make the following observations:

1. From the Topic Coherence results, given different number of topics, our framework consistently achieve higher Topic Coherence scores than the baseline models. Among them, GK-LDA performs best with the highest Topic Coherence score. GK-LDA and MDK-LDA improves significantly ($p < 0.001$) over the three baseline models and MDK-LDA(b).
2. From the KL-Divergence results, given different number of topics, GK-LDA produces the most distinctive topics with the largest KL-Divergence. GK-LDA improves significantly over

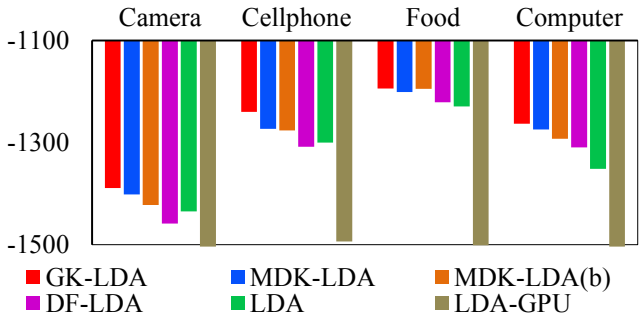


Figure 4. Detailed Average Topic Coherence score of $T = 15$.

The models of each bar from left to rights are GK-LDA, MDK-LDA, MDK-LDA(b), DF-LDA, LDA, and LDA-GPU.

DF-LDA ($p < 0.03$) and all other models ($p < 0.001$). Note that GK-LDA performs much better than MDK-LDA in terms of KL-Divergence.

3. Although DF-LDA has larger KL-Divergence than MDK-LDA and LDA, its Topic Coherence score is not as high as MDK-LDA and LDA. The wrong knowledge does degrade the performance of DF-LDA due to its incapability of handling wrong knowledge. In Section 4.2.2, we further analyze the effects of knowledge on DF-LDA in more details.
4. LDA-GPU does not produce as good topics as other models with the lowest Topic Coherence score and smallest KL-Divergence. As frequent words usually have high co-document frequency with many other words, the frequent words are ranked top in many topics. This shows that general lexical knowledge is more effective than co-document frequency without knowledge as was proposed in [30].
5. In general, with more topics, the Topic Coherence score increases while KL-Divergence decreases, which is in accordance with results in [21, 32]. We found that when T is larger than 15, topics became more and more similar with each other (average KL-Divergence < 2.5). In addition, 15 topics of each model for four domains are a reasonable amount of work for human evaluation. Thus, we fix $T = 15$ to compare the detailed Topic Coherence results (in Section 4.2.2) and human evaluation results (in Section 4.3).

4.2.2 Effects of Knowledge

In order to see the effects and sensitivity to general lexical knowledge, we show the detailed Topic Coherence score of $T = 15$ in Figure 4. Again, we can find that the GK-LDA model has the highest scores across four domains, meaning that it produces the most coherent topics. We can also see that DF-LDA performs better than LDA in the domain "Food" and "Computer" but worse in the domain "Camera" and "Cellphone". In order to fully understand it, we investigated the knowledge in each domain. We found that the knowledge is very different in the four domains:

1. In the domains "Food" and "Computer", the knowledge is simpler with one word usually expressing one meaning/sense. Also, most of the wrong pieces of knowledge in these two domains only contain infrequent words. For example, {*menu, bill*} is not a suitable knowledge for the domain "Computer". But since both *menu* and *bill* are very rare words in this domain, their probabilities under each topic are already very low (< 0.0001). Even if DF-LDA makes their probability closer, the redistribution of probability mass does not influence the probability of frequent (or important) words. Thus, the benefits of correct knowledge outweigh the costs of wrong knowledge and hence DF-LDA performs better than LDA in the domains "Food" and "Computer". However, since there is still a small amount of knowledge involving multiple senses (and may be wrong) which is harmful to DF-LDA, it is not performing as well as our framework.
2. On the other hand, in the domains "Camera" and "Cellphone", the knowledge is more complicated with the words having multiple senses (resulting in wrong knowledge) and mixing of frequent and infrequent words. In this case, DF-LDA performs worse than LDA due to its inability to deal with them.

Table 3. Cohen's Kappa for pairwise inter-rater agreements.

	Topic Labeling	Word Labeling			
		p@5	*p@10*	*p@15*	*p@20*
Kappa	0.915	0.844	0.881	0.916	0.895

In summary, we can conclude that our proposed framework is highly effective in producing distinctive topics where each topic is highly coherent compared to the baseline models. Note that by no means do we say that LDA-GPU and DF-LDA are not effective. We are only saying that in our problem setting of using general lexical knowledge, these models do not generate as coherent topics as ours because they cannot effectively deal with wrong knowledge, which is an important issue.

4.3 Human Evaluation

4.3.1 Quantitative Results

Since our aim is to make topics more interpretable and conform to human knowledge, we worked with two judges who are familiar with Amazon products and reviews to evaluate the models subjectively. Since topics from topic models are rankings based on word probability and we do not know the number of correct topical words, a natural way to evaluate these rankings is to use *Precision @ n* (or *p@n*) which was also used in [31, 43], where *n* is the rank position. We give *p@n* for *n* = 5, 10, 15 and 20. There are two steps in human evaluation: Topic labeling and Word labeling.

Topic Labeling: We followed the instructions in [30] and asked the judges to label each topic as *good* or *bad*. Each topic was presented as a list of 20 most probable words in descending order of their probabilities under that topic. The models which generated the topics for labeling were oblivious to the judges. In general, each topic was annotated as *good* if it had more than half of its words coherently related to each other representing a semantic concept together; otherwise *bad*. Table 3 (column 2) reports the Cohen's Kappa score for topic labeling, which is above 0.8 indicating almost perfect agreements according to scale in [22].

Word Labeling: After topic labeling, we chose the topics, which were labeled as good by both judges, as good topics. Then, we asked the two judges to label each word of the top 20 words in these good topics. Each word was annotated as *correct* if it was coherently related to the concept represented by the topic; otherwise *incorrect*. Since judges already had the conception of each topic in mind when they were labeling topics, labeling each word was not very difficult which explains the high Kappa scores in Table 3 (column 3). We can see that both annotators achieve almost perfect agreements (Kappa > 0.8) in all *p@n*.

Precision @ n: Figure 5 gives the average *precision @ n* of all good topics over all four domains. We can make the following observations:

1. GK-LDA performs the best, improving LDA by more than 11% on average. The model successfully identifies and leverages the correct knowledge and also addresses the wrong knowledge in the form of LR-sets for each domain.

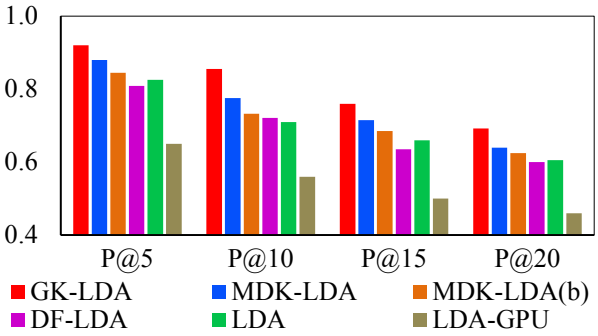


Figure 5. Average *Precision @ n* (*p @ n*) of good topics over all four domains.

The models of each bar from left to rights are GK-LDA, MDK-LDA, MDK-LDA(b), DF-LDA, LDA, and LDA-GPU. (Same for Figure 6)

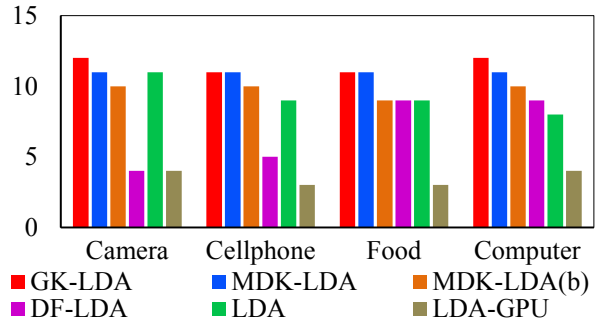


Figure 6. Number of good topics generated by each model.

2. MDK-LDA improves precision of MDK-LDA(b) by about 3% and LDA by about 5%. We can see that the promotion of LR-sets through GPU model is effective. But it still suffers from the wrong knowledge. MDK-LDA(b) improves the precision of LDA by only 2%.
3. DF-LDA performs slightly worse than LDA with less than 1%. This shows that not dealing with wrong knowledge can dramatically reduce the model's effectiveness, which is understandable because the wrong knowledge may guide the model mistakenly.
4. LDA-GPU does not perform well in our data due to its use of co-document frequency. GK-LDA performs much better. Again, this shows that LR-set knowledge is more effective than co-document frequency.

We can see that the human evaluation results are highly consistent with Topic Coherence and KL-Divergence results in Section 4.2. Upon significance testing of improvement of GK-LDA and MDK-LDA over other models in Figure 5, GK-LDA and MDK-LDA improves significantly over all other models ($p < 0.005$).

Number of Good topics: Figure 6 shows the number of good topics discovered by each model in each domain. In general, GK-LDA can generate about 2 more good topics than LDA and more than 5 additional topics compared to DF-LDA and LDA-GPU. These are very important in practice. For DF-LDA, it discovered fewer good topics than LDA in the domains "Camera" and "Cellphone" but more good topics than LDA in the domain "Computer", which is consistent with the analysis in Section 4.2.2. We also found that all topics discovered by LDA, LDA-GPU and DF-LDA can be uncovered by MDK-LDA(b), MDK-LDA and GK-LDA. Thus, our framework not only produces additional good topics but also keeps the good topics of the baseline models.

4.3.2 Qualitative Results

This section shows some qualitative results to give an intuitive feeling of the results from different models. There are a large number of topics that GK-LDA makes major improvements. Due to space limitations, we can only show some examples. To further focus, we will just show some results of LDA and GK-LDA. The results from LDA-GPU and DF-LDA were inferior and even hard to match with topics found by the other models.

Table 4 shows 6 example topics and top ranked topical words from LDA and GK-LDA. Wrong topical words are in italic and marked red (we tried to find the best possible match for the models). We can see that GK-LDA produces much better topics. Since the labeling of topics and topical words are somewhat subjective, we do not expect everyone to agree with the labeling, but we tried our best to have the consensus with two human judges. Clearly, the results in Table 4 do not tell all the story. We also want to highlight several important points below.

- One of the most common and important topics in online reviews is the *price* of products. However, out of the four domains, only in the domain "Food", LDA was able to find the topic *price* with a reasonable precision. In other domains, the price related topical words were mixed with all kinds of other topics by LDA. We show one example in Table 4 (in the column "Cellphone"), where the best *price* related topic of LDA is still poor. We believe that LDA's inability is mainly due to the fact that in English, sentences like "*The price of this phone is expensive*." are relatively rare. Thus, there is probably no co-occurrence of *price* and *expensive* (or other adjectives related to *price*, e.g., *cheap*) within a sentence. Our adjective-attribute knowledge is very effective in this case, discovering the good *price* topic (see "GK-LDA" in the column "Cellphone").
- LDA tends to split one topic into multiple topics, i.e., the topical words for a semantic topic appear at the top ranked positions of several topics. GK-LDA is much better in this regard. The results in Table 4 also show that.
- There are also many other examples we could not list here due to space limitations. For example, in the food domain, GK-LDA discovered the topic *Healthy Eating*: *protein*, *fat*, *fiber*, *healthy, nutrient, nutrition*, *vitamin*, *magnesium*. These words are all highly coherent. The best one that LDA could find were: *time*, *snack*, *food*, *point*, *healthy*, *calorie*, *weight*, *year*. In the computer domain, GK-LDA was able to find the topic of *Program Execution*: *game*, *slow*, *word*, *fast*, *web*, *star*, *speed*, *slower*. Most of the words here are highly relevant except *star*. LDA was unable to find any topic related.

In summary, we can say that GK-LDA produces much better results, both in terms of precision and the number of good topics, which indicates that our proposed framework of exploiting general lexical knowledge is highly promising.

5. CONCLUSIONS

This paper proposed a novel task of utilizing the general knowledge of lexical semantic relations in topic models in order to produce more coherent topics. In any language, there is a vast amount of such knowledge stored in dictionaries. Since such knowledge is domain independent, it should be applicable to any application domain. However, due to multiple meanings or senses of a word, some knowledge may not be suitable for a particular application domain. This paper proposed the GK-LDA model as a comprehensive framework to effectively leverage general knowledge in topic models and to also deal with the wrong knowledge. To our knowledge, this is the first work that proposes

Table 4. Six example topics in four domains (Errors are marked in red/italic).

Camera		Cellphone		Food				Computer			
Photographer		Price		Family		Taste		Usage		Image Quality	
LDA	GK-LDA	LDA	GK-LDA	LDA	GK-LDA	LDA	GK-LDA	LDA	GK-LDA	LDA	GK-LDA
dslr	dslr	*product*	price	*bag*	kid	taste	salt	*acer*	easy	quality	resolution
point	*year*	price	cheap	*package*	family	salt	taste	*power*	simple	picture	pixel
year	professional	*review*	money	*waffle*	husband	*almond*	flavor	*base*	control	*easy*	quality
canon	amateur	*time*	expensive	*microwave*	daughter	*fresh*	tasty	*year*	difficult	high	image
photography	pro	*item*	cost	*love*	baby	*pack*	delicious	button	setting	*money*	picture
nikon	photography	*device*	cheaper	baby	wife	tasty	sweet	*amazon*	hard	*inch*	*dead*
photographer	experience	money	inexpensive	family	child	*oil*	salty	control	easier	movie	high
shoot	*month*	*star*	*shipping*	husband	son	*roasted*	tasting	*price*	instruction	*price*	low
price	photographer	cheap	worth	*dish*	*jelly*	pepper	spice	*color*	manual	*problem*	higher
digital	*model*	*shipping*	dollar	kid	*snack*	*easy*	yummy	*purchase*	*software*	*size*	lower

a principled model to systematically incorporate the general knowledge to produce more coherent topics. What is even more important is that our proposed framework can automatically deal with wrong knowledge without any user input. Given the success of topic models in many research areas, we feel that our proposed framework for encoding general knowledge presents a promising direction to advance the current state-of-the-art in the field.

6. ACKNOWLEDGMENTS

This work was supported in part by a grant from National Science Foundation (NSF) under grant no. IIS-1111092, and a grant from HP Labs Innovation Research Program.

7. REFERENCES

[1] Andrzejewski, D., Zhu, X. and Craven, M. 2009. Incorporating domain knowledge into topic modeling via Dirichlet Forest priors. *ICML*, 25–32.

[2] Andrzejewski, D., Zhu, X., Craven, M. and Recht, B. 2011. A framework for incorporating general domain knowledge into latent Dirichlet allocation using first-order logic. *IJCAI*, 1171–1177.

[3] Barker, C. 2006. *Lexical Semantics*. Encyclopedia of Cognitive Science.

[4] Blei, D.M. and McAuliffe, J.D. 2007. Supervised Topic Models. *NIPS*, 121–128.

[5] Blei, D.M., Ng, A.Y. and Jordan, M.I. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research*. 3, 993–1022.

[6] Boyd-Graber, J.L., Blei, D.M. and Zhu, X. 2007. A Topic Model for Word Sense Disambiguation. *EMNLP-CoNLL*, 1024–1033.

[7] Brody, S. and Elhadad, N. 2010. An unsupervised aspect-sentiment model for online reviews. *NAACL*, 804–812.

[8] Burns, N., Bi, Y., Wang, H. and Anderson, T. 2012. Extended Twofold-LDA Model for Two Aspects in One Sentence. *Advances in Computational Intelligence*. Springer Berlin Heidelberg. 265–275.

[9] Chang, J., Boyd-Graber, J., Chong, W., Gerrish, S. and Blei, M., D. 2009. Reading Tea Leaves: How Humans Interpret Topic Models. *NIPS*, 288–296.

[10] Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M. and Ghosh, R. 2013. Leveraging Multi-Domain Prior Knowledge in Topic Models. *IJCAI*, 2071–2077.

[11] Fei, G., Liu, B., Hsu, M., Castellanos, M. and Ghosh, R. 2012. A Dictionary-Based Approach to Identifying Aspects Implied by Adjectives for Opinion Mining. *COLING (Posters)*, 309–318.

[12] Griffiths, T.L. and Steyvers, M. 2004. Finding Scientific Topics. *PNAS*. 101 Suppl, 5228–5235.

[13] Hartung, M. and Frank, A. 2011. Exploring supervised lda models for assigning attributes to adjective-noun phrases. *EMNLP*, 540–551.

[14] Heinrich, G. 2009. A Generic Approach to Topic Models. *ECML PKDD*, 517 – 532.

[15] Hofmann, T. 1999. Probabilistic Latent Semantic Analysis. *UAI*, 289–296.

[16] Hong, L., Yin, D., Guo, J. and Davison, B.D. 2011. Tracking trends: incorporating term volume into temporal topic models. *KDD*, 484–492.

[17] Hu, Y., Boyd-Graber, J. and Satinoff, B. 2011. Interactive Topic Modeling. *ACL*, 248–257.

[18] Jagarlamudi, J., III, H.D. and Udupa, R. 2012. Incorporating Lexical Priors into Topic Models. *EACL*, 204–213.

[19] Jo, Y. and Oh, A.H. 2011. Aspect and sentiment unification model for online review analysis. *WSDM*, 815–824.

[20] Kataria, S., Mitra, P., Caragea, C. and Giles, C.L. 2011. Context Sensitive Topic Models for Author Influence in Document Networks. *IJCAI*, 2274–2280.

[21] Kawamae, N. 2010. Latent interest-topic model. *CIKM*, 649–658.

[22] Landis, J. and Koch, G. 1977. The measurement of observer agreement for categorical data. *biometrics*. 33.

[23] Li, L., Roth, B. and Sporleder, C. 2010. Topic models for word sense disambiguation and token-based idiom detection. *ACL*, 1138–1147.

[24] Lin, C., Yang, J.-M., Cai, R., Wang, X.-J., Wang, W. and Zhang, L. 2009. Simultaneously modeling semantics and structure of threaded discussions: a sparse coding approach and its applications. *SIGIR*, 131–138.

[25] Lu, B., Ott, M., Cardie, C. and Tsou, B.K. 2011. Multi-aspect Sentiment Analysis with Topic Models. *ICDM Workshops*, 81–88.

[26] Lu, Y. and Zhai, C. 2008. Opinion integration through semi-supervised topic modeling. *WWW*, 121–130.

[27] Mahmoud, H. 2008. *Polya Urn Models*. Chapman & Hall/CRC Texts in Statistical Science.

[28] Mei, Q., Cai, D., Zhang, D. and Zhai, C. 2008. Topic modeling with network regularization. *WWW*, 101–110.

[29] Miller, G.A. 1995. WordNet: A Lexical Database for English. *Commun. ACM*. 38, 11, 39–41.

[30] Mimno, D., Wallach, H.M., Talley, E., Leenders, M. and McCallum, A. 2011. Optimizing semantic coherence in topic models. *EMNLP*, 262–272.

[31] Mukherjee, A. and Liu, B. 2012. Aspect Extraction through Semi-Supervised Modeling. *ACL*, 339–348.

[32] Mukherjee, A. and Liu, B. 2012. Mining contentions from discussions and debates. *KDD*, 841–849.

[33] Newman, D., Bonilla, E. V. and Buntine, W.L. 2011. Improving Topic Coherence with Regularized Topic Models. *NIPS*, 496–504.

[34] Newman, D., Noh, Y., Talley, E., Karimi, S. and Baldwin, T. 2010. Evaluating topic models for digital libraries. *JCDL*, 215–224.

[35] Petterson, J., Smola, A., Caetano, T., Buntine, W. and Narayanamurthy, S. 2010. Word Features for Latent Dirichlet Allocation. *NIPS*, 1921–1929.

[36] Ramage, D., Hall, D., Nallapati, R. and Manning, C.D. 2009. Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. *EMNLP*, 248–256.

[37] Robert, C.P. and Casella, G. 2004. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc. Secaucus, NJ, USA.

[38] Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P. and Steyvers, M. 2010. Learning author-topic models from text corpora. *ACM Transactions on Information Systems*. 28, 1, 1–38.

[39] Stevens, K. and Buttler, P. 2012. Exploring Topic Coherence over many models and many topics. *EMNLP-CoNLL*, 952–961.

[40] Teh, Y.W., Jordan, M.I., Beal, M.J. and Blei, D.M. 2006. Hierarchical dirichlet processes. *Journal of the American Statistical Association*. 1–30.

[41] Titov, I. and McDonald, R. 2008. Modeling online reviews with multi-grain topic models. *WWW*, 111–120.

[42] Zhang, D., Mei, Q. and Zhai, C. 2010. Cross-Lingual Latent Topic Extraction. *ACL*, 1128–1137.

[43] Zhao, W.X., Jiang, J., Yan, H. and Li, X. 2010. Jointly Modeling Aspects and Opinions with a MaxEnt-LDA Hybrid. *EMNLP*, 56–65.