Engineering Applications of Artificial Intelligence 64 (2017) 432–439

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Mining coherent topics in documents using word embeddings and large-scale text data

Liang Yao, Yin Zhang *, Qinfei Chen, Hongze Qian, Baogang Wei, Zhifeng Hu

College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Keywords:
Topic model
Domain knowledge
Word embedding
Large-scale text data

ABSTRACT

Probabilistic topic models have been extensively used to extract low-dimension aspects from document collections. However, such models without any human knowledge often generate topics that are not interpretable. Recently, a number of knowledge-based topic models have been proposed, which enable users to input prior domain knowledge to produce more meaningful and coherent topics. Word embeddings, on the other hand, can automatically capture both semantic and syntactic information of words from a large amount of documents, and can be used to measure word similarities. In this paper, we incorporate word embeddings obtained from a large number of domains into topic modeling. By combining Latent Dirichlet Allocation, a widely used topic model with Skip-Gram, a well-known framework for learning word vectors, we improve the semantic coherence significantly. Our evaluation results using product review documents from 100 domains will demonstrate the effectiveness of our method.

1. Introduction

The explosive growth of online text content, such as Twitter messages, blogs, news and product reviews has brought about the challenge to understand the very dynamic sea of text. To deal with the challenge, we need to discover concepts from massive text.

A number of text mining tasks, especially aspects extraction tasks, utilize probabilistic topic models such as Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA) (Hofmann, 1999; Blei et al., 2003). However, these unsupervised models without any human knowledge often result in topics that are difficult to interpret. In other words, they could not produce semantically coherent concepts (Chang et al., 2009; Mimno et al., 2011).

To overcome the shortcoming of interpretability in topic models, especially in LDA, some previous works incorporate prior domain knowledge into topic modeling in different ways. However, they either cannot learn knowledge automatically, or fail to utilize multiple domain data sufficiently.

Topic models such as LDA utilize the bag of word representation and document-level word co-occurrence to assign a topic to each word observation in the corpus. Similarly, word embeddings (Bengio et al., 2003; Mnih and Hinton, 2007; Collobert and Weston, 2008; Collobert et al., 2011; Huang et al., 2012; Mikolov et al., 2013) conduct dimensionality reduction based on co-occurrence information, but focus more on local context and word order in text to learn a low-dimension dense word vector for each word. Word embeddings aim at explicitly encoding many semantic relationships as well as linguistic regularities and patterns into new embedding space. For example, the result of a vector calculation vec("Madrid") − vec("Spain") + vec("France") is closer to vec("Paris") than to any other word vector, and "Spain" is close to "France" in the embedding space. Since similar words are close in embedding space, we can utilize the word correlation knowledge encoded by word embeddings.

In this paper, we improve previous knowledge-based topic models by proposing a new probabilistic method, called Word Embedding LDA (WE-LDA), which combines topic model and word embeddings, in particular LDA model and Skip-Gram (Mikolov et al., 2013). The proposed method explicitly models document-level word co-occurrence in the corpus with word correlation knowledge encoded by word vectors automatically learned from a large amount of relevant data, which could extract more coherent topics in documents.

The contributions of the paper are threefold: (1) It proposes a novel knowledge mining method for topic modeling based on word embeddings. (2) It provides a novel knowledge-based topic model which could handle the knowledge encoded by word embeddings properly.

* Corresponding author.
E-mail addresses: yaoliang@zju.edu.cn (L. Yao), yinzh@zju.edu.cn (Y. Zhang), chenqinfei@zju.edu.cn (Q. Chen), azureqianhz@zju.edu.cn (H. Qian), wbg@zju.edu.cn (B. Wei), huyangc@zju.edu.cn (Z. Hu).

http://dx.doi.org/10.1016/j.engappai.2017.06.024
Received 14 June 2016; Received in revised form 27 January 2017; Accepted 26 June 2017
0952-1976/

(3) Comprehensive experimental results on two large e-commerce domain datasets demonstrate our method outperforms six state-of-the-art knowledge-based topic models.

We begin this paper by introducing some related works, including studies which devote to improving the semantic coherence of topic models mainly by incorporating domain knowledge into topic models, studies which measure the coherence of topic models, and studies which focus on learning word representation. In the remainder of this paper, we first describe our model, then empirically evaluate our method on real world datasets and analyze experimental results. Experiments on two large product review datasets show the effectiveness of our method.

2. Related work

2.1. Knowledge-based topic models

To overcome the drawback of interpretability in topic models, especially in LDA, some previous works incorporate prior domain knowledge into topic modeling. For instance, Andrzejewski and Zhu (2009) proposed topic-in-set knowledge which restricts topic assignment of words to a subset of topics. Andrzejewski et al. (2011) extended topic-in-set knowledge (Andrzejewski and Zhu, 2009) by incorporating general knowledge specified by first-order logic. Similarly, Chemudugunta et al. (2008) proposed Concept model by utilizing ontologies like Open Directory Project (ODP) or The Cambridge International Dictionary of English (CIDE). The DF-LDA (Dirichlet Forest LDA) model in Andrzejewski et al. (2009) could incorporate knowledge in the form of must-links and cannot-links input by users. A must-link states that two words should share the same topic, while a cannot-link means two words should not be in the same topic. Newman et al. (2011) proposed two Bayesian regularization formulations to improve topic coherence. Both methods use additional word co-occurrence data to improve the coherence and interpretability of learned topics. Hu et al. (2011) developed a framework for allowing users to iteratively refine the topics by adding constraints that enforce that sets of words must appear together in the same topic.

Recently, LDA with Multi-Domain Knowledge (MDK-LDA) (Chen et al., 2013c) was presented, MDK-LDA is capable of using prior knowledge from multiple domains. In Chen et al. (2013b), a knowledge-based topic model, called MC-LDA (LDA with must-link set and cannot-link set), was proposed as an extension of MDK-LDA. MC-LDA assumes that all knowledge is correct and uses must-link and cannot-link knowledge as DF-LDA. GK-LDA (General Knowledge based LDA) is another knowledge-based topic model that uses the ratio of word probabilities under each topic to reduce the effect of wrong knowledge (Chen et al., 2013a). More recently, Probase-LDA (Yao et al., 2015) and Concept over Time (Yao et al., 2016), a method that combines topic model and Probase (a probabilistic knowledge base) and a method that combines topic model and Wikipedia knowledge, were put forward. The methods can model text content with the consideration of probabilistic knowledge or encyclopedia knowledge for detecting better topics. Yang et al. (2015) explored leveraging existing prior knowledge into topic modeling when dealing with large datasets.

Although above mentioned knowledge-based topic models utilize knowledge in many ways, they only process human input knowledge, but could not learn knowledge automatically.

To address the issue, AKL (Automated Knowledge LDA), LTM (Lifelong Topic model) and AMC (topic modeling with Automatically generated Must-links and Cannot-links) were proposed (Chen et al., 2014; Chen and Liu, 2014b, a), which learn knowledge automatically from multiple domains to improve topics in each domain. Our work is closely related to these methods. Despite the fact that these methods are effective, they only use frequent itemset mining to mine knowledge from top topical words in multiple domains, and do not consider the order of words in text, our method, on the other hand, utilizes multiple domain data more sufficiently by exploiting word vectors.

In order to measure the interpretability of topic models, Mimno et al. (2011) introduced an automatic coherence measure using word co-occurrence in the training corpus, which automates the human judgment approach in Chang et al. (2009). At the same time, they proposed an unsupervised generalized Pólya urn (GPU) method which improves coherence score by considering the word co-document frequency in the corpus. Newman et al. (2010) showed that an automated evaluation metric based on word co-occurrence statistics gathered from Wikipedia can reflect human evaluations of topic quality. Chuang et al. (2013) measured the correspondence between a set of latent topics and a set of reference concepts when applying topic models to domain-specific tasks, which gives another way to appraise the coherence. Word embedding has also been used to evaluate the coherence of topics from Twitter data recently Fang et al. (2016).

2.2. Word embeddings

Since the innovative work of the neural network language model (Bengio et al., 2003), a number of studies (Mnih and Hinton, 2007; Collobert and Weston, 2008; Collobert et al., 2011; Huang et al., 2012) have devoted to building distributed word representations. The dense real-valued vector representations capture local context information of words and represent their "meaning", where the meaning of a word is determined by its surrounding words. Recently, the efficient continuous bag of words (CBOW) model and the continuous Skip-gram model (Mikolov et al., 2013) have been proposed. The training objective of CBOW is to combine the embeddings of surrounding words to predict the central word in a sliding window; while Skip-Gram tries to use the central word as input to predict the surrounding words in a sliding window.

Serving as invaluable features, word vectors have been used in NLP applications such as Part-Of-Speech tagging, chunking, named entity recognition and synonym detection (Collobert et al., 2011; Baroni et al., 2014). Word embeddings are useful because they can encode both syntactic and semantic information of words into continuous vectors and similar words are close in vector space.

Some previous studies have used word embeddings to encode semantic regularities. Nguyen et al. (2015) extended topic models by incorporating latent feature vectors of words learned from very large corpora. Das et al. (2015) replaced LDA's parameterization of "topics" as multinomial distributions over words with multivariate Gaussian distributions on the word embedding space. However, they consider all word vectors under a topic and some less related words of a target word may result in some noise in the learning process. Moreover, learning latent feature vector and estimating parameters of multivariate Gaussian distributions are complex. Our method, on the other hand, simply uses the most related words of a target word to generate word correlation knowledge and affect the sufficient statistics directly, which is easier for model inference.

3. The WE-LDA model

The proposed WE-LDA model consists of three steps. First, we run LDA and select topical words as seed words of a corpus. Then we use word vectors to generate the must-link knowledge base. Finally, we take the generalized Pólya urn (GPU) method (Mahmoud, 2008; Mimno et al., 2011) which is the key technique for incorporating must-links into Gibbs sampling, and find more semantically coherent topics. The first and the second step aim to generate high quality prior knowledge for topic modeling. The third step is to use the learned knowledge.

3.1. Seed words generation

Since the topics found by LDA are an intuitive summary of a corpus, and top words under each topic are more likely to represent the main semantics of the corpus, we use top n words with highest probability under each topic in T topics as seed words, to further generate the must-link knowledge base of a corpus. We select top n words of each topic from LDA instead of frequent words because we want to keep the diversity of the seed words. Words in different topics are often different and some top topical words may not be the frequent ones. Formally, the top n words set W_t under a topic t is defined as:

$$W_t = n\text{-argmax}_w \phi_t(w) \tag{1}$$

where w is a word in the vocabulary of the corpus, $\phi_t(w)$ is the probability of w under t estimated by LDA, here we use the n-argmax operator to yield the n arguments which result in the n largest values for the given function.

3.2. Knowledge mining

As word embedding can encode the semantic relationship between two words, we train Skip-Gram (Mikolov et al., 2013) on large-scale (multiple domains) text to generate word vectors which encode general domain knowledge. Specifically, Skip-Gram is defined as follows: Given a word sequence $w_1, w_2, \dots, w_X$, the objective of the Skip-Gram model is to maximize the following average log probability:

$$L = \frac{1}{X}\sum_{x=1}^{X}\sum_{-N\le c\le N, c\neq 0} \log p(w_{x+c}|w_x) \tag{2}$$

where w_x is the central word, w_{x+c} is a surrounding word, and N indicates the context window size to be $2N+1$. The conditional probability $p(w_{x+c}|w_x)$ is calculated using a softmax function as follows:

$$p(w_{x+c}|w_x) = \frac{\exp(\mathbf{v}_{w_{x+c}} \cdot \mathbf{v}_{w_x})}{\sum_{w=1}^{V}\exp(\mathbf{v}_w \cdot \mathbf{v}_{w_x})} \tag{3}$$

where $\mathbf{v}_{w_x}$ and $\mathbf{v}_{w_{x+c}}$ are the vector representations of the w_x and w_{x+c} respectively, and V is the vocabulary size. The average log probability is optimized by using stochastic gradient descent. Updates for $\mathbf{v}_{w_{x+c}}$ and $\mathbf{v}_{w_x}$ are:

$$\mathbf{v}_{w_{x+c}} - \alpha_s\left(\frac{\exp\{f(w_p)\}}{\exp\{f(w_p)\}+1} - I[w_{x+c}=w_p]\right)\mathbf{v}_{w_x} \tag{4}$$

$$\mathbf{v}_{w_x} - \alpha_s \sum_{-N\le c\le N, c\neq 0}\left(\frac{\exp\{f(w_p)\}}{\exp\{f(w_p)\}+1} - I[w_{x+c}=w_p]\right)\mathbf{v}_{w_{x+c}} \tag{5}$$

where w_p is the central word predicted by the model, $I[x]$ is 1 when x is true, $f(w_p) = \mathbf{v}_{w_{x+c}} \cdot \mathbf{v}_{w_x}$, α_s is learning rate.

Then for each word in the seed words set, we use the cosine similarity of word vectors trained by Skip-Gram to measure the semantic similarity between the word and other words. We use each word in the seed words set with its top m similar words in multiple domains text to generate the knowledge base. The cosine similarity of two words w_1 and w_2 is defined as:

$$sim(w_1, w_2) = \frac{\vec{v}_{w_1}\vec{v}_{w_2}}{|\vec{v}_{w_1}||\vec{v}_{w_1}|} \tag{6}$$

where $\vec{v}_w$ is the word embedding of w, the set of w's top m similar words $Must_w$ is defined as:

$$Must_w = m\text{-argmax}_{w'} sim(w, w'). \tag{7}$$

The similarity of word embeddings is essentially the similarity of word context. We treat a word and one of its similar words as a must-link, which has been proved to be an effective prior knowledge form for topic modeling (Andrzejewski et al., 2009), and use the combination of cosine similarity and the Point-wise Mutual Information (PMI), a popular measure of words association in text, to measure the semantic relatedness of a must-link. PMI can be interpreted as how likely two words co-occur in a document. The use of PMI is to overcome the issue of must-links: our must-links are generated using multiple domains text and some of them may not be suitable for current domain, and PMI only considers words association in current corpus (domain), we want to use must-links which carry both general domain knowledge and domain-specific knowledge. The PMI value is computed as:

$$PMI(w_1, w_2) = \log\frac{P(w_1, w_2)}{P(w_1)P(w_2)} \tag{8}$$

$$P(w) = \frac{\#D(w)}{\#D} \tag{9}$$

$$P(w_1, w_2) = \frac{\#D(w_1, w_2)}{\#D} \tag{10}$$

where $\#D(w)$ is the number of documents in a corpus that contain the word w, $\#D(w_1, w_2)$ is the number of documents that contain both words w_1 and w_2, and $\#D$ is the total number of documents in the corpus. A positive PMI value indicates a high semantic correlation of words in current corpus (domain), while a non-positive PMI value implies little or no semantic correlation in current corpus (domain). Therefore, we only consider must-links with positive PMI values.

Consequently, the word relatedness of w_1 and w_2 is computed as:

$$\lambda_{w_1,w_2} = \begin{cases} 1, & w_1 = w_2 \\ \mu \times r(w_1, w_2), & (w_1, w_2) \text{ is a must-link} \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

where μ is a parameter factor that controls how much the WE-LDA model should trust the word relatedness, $r(w_1, w_2) = PMI(w_1, w_2) \times sim(w_1, w_2)$ is the product of PMI value and word vector similarity.

Some example must-links with their similarity and PMI values in our experiment are listed in Table 1.

3.3. Gibbs sampler

After the generation of the must-link knowledge base, the problem is how to incorporate must-links into topic modeling. The key is we need to estimate how much a must-link is related to a certain topic. Since similar words tend to be in the same topic, in this part, we simply group all must-links into T groups using k-means clustering,[1] where T is the number of topics. When performing k-means clustering, we use the sum of word vectors of the two words in a must-link as the feature vector of the must-link. Each must-link will be assigned to a topic k, and we can note the relatedness λ_{w_1,w_2} as λ_{k,w_1,w_2}.

After grouping must-links, we finally take the GPU approach of Mimno et al. (2011) which approximates the true Gibbs sampling distribution by treating each word as if it were the last. GPU is designed in the context of colored balls in a urn. It means seeing a ball of a certain color will not only increase the probability of the color, but also increase the probability of seeing balls with other similar colors. The GPU model in our context basically states that assigning topic z_i to word w_i will not only increase the probability of connecting z_i with w_i, but also make it more likely to associate z_i with word w' where w' shares a must-link with w_i under the topic (must-link group) z_i. We can use the word relatedness $\lambda_{k,w_i,w'}$ as the weight of topic-word counter, the topic-word counter is a sufficient statistic. The approximate Gibbs sampler is based on the following conditional distribution: $P(z_i = k|z_{-i}, \alpha, \beta, \lambda) \propto$

$$\frac{n_{dk}+\alpha}{n_d + T\alpha} \times \frac{\sum_{w'=1}^{V}\lambda_{k,w_i,w'} \times n_{kw'} + \beta}{\sum_{v=1}^{V}\left(\sum_{w'=1}^{V}\lambda_{k,v,w'} \times n_{kw'} + \beta\right)} \tag{12}$$

where z_i is the topic assignment of current word w_i. n_{dk} is the number of times topic k is assigned to a word in current document d. n_d is the length of document d. $n_{kw'}$ is topic-word counter, i.e., the number of times w' is

[1] We tried k-means and k-medoids, and found k-means performs slightly better.

Table 1
Some example must-links.

w_1	w_2	$sim(w_1, w_2)$	$PMI(w_1, w_2)$	$r(w_1, w_2) = PMI(w_1, w_2) \times sim(w_1, w_2)$
Review	Comment	0.729	2.256	1.645
Review	Positive	0.635	3.146	1.997
Review	Reviewer	0.587	1.750	1.027
Warranty	Repair	0.820	3.153	2.585
Warranty	Replacement	0.746	2.342	1.747
Warranty	Manufacturer	0.695	2.794	1.942
Windows	xp	0.762	4.789	3.649
Windows	linux	0.705	3.019	2.128
Windows	os	0.699	4.657	3.255

assigned to topic k. α and β are predefined Dirichlet hyper-parameters. $\lambda_{k,w_i,w'}$ is the word relatedness of a must-link w_i and w' which has been grouped into group k. T is the number of topics, and V is the vocabulary size.

4. Experimental results

This section evaluates the proposed WE-LDA model and compares it with seven state-of-the-art baseline models:

- **LDA** (Blei et al., 2003): A classic unsupervised topic model.
- **LDA-GPU** (Mimno et al., 2011): LDA with GPU, an unsupervised topic model. Specifically, LDA-GPU applies GPU in LDA using co-document frequency.
- **GK-LDA** (Chen et al., 2013a): A knowledge-based topic model. It uses the ratio of word probabilities under each topic to reduce the effect of wrong knowledge.[2]
- **LTM** (Chen and Liu, 2014b): A lifelong learning topic model. It learns the must-link type of knowledge automatically.[3] It outperforms AKL (Chen et al., 2014).
- **AMC** (Chen and Liu, 2014a): A lifelong learning topic model that learns the must-link and cannot-link type of knowledge automatically.[4]
- **Probase-LDA** (Yao et al., 2015): A knowledge-based topic model that leverages a probabilistic knowledge base to calculate the priors of topic models.[5]
- **LF-LDA** (Nguyen et al., 2015), a knowledge-based topic model that extends LDA by incorporating word embeddings learned from very large corpora.[6]

4.1. Datasets and settings

Datasets. We use two large datasets from Chen and Liu (2014a). The first dataset contains reviews from 50 types of electronic products or domains. The second dataset contains reviews from 50 types of non-electronic products or domains. Each domain has 1,000 reviews. The dataset has been pre-processed by the authors, each review has been divided into sentences, and each sentence is treated as a document.

Settings. For comparison, we use the same parameter settings as Chen and Liu (2014b, a) and Yao et al. (2015). All models except LF-LDA are trained using 2,000 iterations with an initial burn-in of 200 iterations. The sampling lag is 20. The hyper-parameters of all models except LF-LDA are set as $\alpha = 1$, $\beta = 0.1$. Other hyper-parameters for baseline models are set as suggested in their original papers. For LF-LDA, we use the default settings in the authors' implementation: https://github.com/datquocnguyen/LFTM. The number of topics for all models is $T = 15$.

[2] https://github.com/czyuan/GKLDA.
[3] https://github.com/czyuan/LTM.
[4] https://github.com/czyuan/AMC.
[5] https://github.com/yao8839836/ProbaseLDA.
[6] https://github.com/datquocnguyen/LFTM.

Table 2
Average topic coherence values of each model with different number of top words on electronic products dataset.

Model	# top words				
	10	15	20	25	30
WE-LDA	−130.10	−306.06	−554.36	−872.56	−1267.19
AMC	−136.37	−326.74	−597.29	−949.53	−1382.31
LTM	−136.56	−335.54	−617.78	−979.40	−1417.24
Probase-LDA	−138.62	−314.45	−558.80	−867.46	−1236.01
GK-LDA	−144.76	−346.25	−631.63	−999.13	−1448.93
LDA	−146.53	−351.17	−640.61	−1012.71	−1465.32
LDA-GPU	−152.16	−371.22	−690.79	−1108.62	−1623.90
LF-LDA	−148.32	−362.55	−668.93	−1061.82	−1536.50

For WE-LDA[7] we use all reviews (50,000) in the 50 domains in each dataset as the input corpus for Skip-Gram. We set the context window size to be 5, the initial learning rate $\alpha_s = 0.025$ and the dimension of word embeddings to be 100 when training Skip-Gram model. Then for each domain, we use top $n = 30$ words under each topic produced by LDA as seed words, and use these seed words with their $m = 20$ most similar words as must-link candidates, but only consider must-links with positive PMI values and set $\mu = 0.8$. For LF-LDA, the input word embeddings are the same as WE-LDA. For LTM, we use test setting 1 in the original paper (mining prior knowledge from topics of all domains in a dataset, including the test domain), we evaluate topics generated by LTM at learning iteration 4 (have stabilized). Since GK-LDA cannot mine any prior knowledge, we feed it the knowledge produced by WE-LDA.

4.2. Topic coherence

We evaluate topics produced by each model based on UMass Topic Coherence (Mimno et al., 2011). Traditionally, topic models have often been evaluated using perplexity on held-out test data. Unfortunately, predictive perplexity does not indicate the interpretability of topics and may be contrary to human evaluation (Chang et al., 2009). Alternatively, the metric Topic Coherence has been shown in Mimno et al. (2011) to correlate well with human judging. Since our objective is to discover coherent or meaningful topics, Topic Coherence is more suitable for our evaluation. The UMass Topic Coherence has also been used in many previous works (Chen and Liu, 2014b, a; Yao et al., 2015; Yang et al., 2015). A higher Topic Coherence value implies a better topic. Topic Coherence is computed as:

$$C(t; V^{(t)}) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} \log \frac{D(v_m^{(t)}, v_l^{(t)}) + 1}{D(v_l^{(t)})} \tag{13}$$

where $D(v)$ is the document frequency of word v (i.e., the number of documents word v appears in), $D(v, v')$ is the co-document frequency of word v and v' (i.e., the number of documents word v and v' both appear in). $V^{(t)} = \{v_1^{(t)}, \ldots, v_M^{(t)}\}$ is the list of M words with highest probabilities in topic t. A smoothing count of 1 is introduced to avoid taking the logarithm of 0.

[7] We released the code at, https://github.com/yao8839836/WE-LDA.

Table 3
Average Topic Coherence values of each model with different number of top words on non-electronic products dataset.

Model	# top words				
	10	15	20	25	30
WE-LDA	−129.34	−302.38	−545.62	−862.29	−1247.16
AMC	−135.14	−319.34	−580.51	−916.20	−1324.60
LTM	−137.50	−329.31	−600.41	−947.66	−1368.70
Probase-LDA	−133.61	−302.15	−531.39	−823.40	−1172.59
GK-LDA	−140.78	−334.55	−611.37	−972.29	−1416.30
LDA	−142.59	−338.76	−616.74	−972.98	−1407.41
LDA-GPU	−155.74	−373.61	−681.05	−1074.96	−1546.94
LF-LDA	−148.33	−351.84	−637.05	−998.65	−1431.69

Table 4
Example topics of WE-LDA, AMC and LDA. Errors are italicized and marked in red.

Microwave (Food Cooking)			Network Adapter (OS & Driver)		
WE-LDA	AMC	LDA	WE-LDA	AMC	LDA
vegetable	food	popcorn	os	vista	driver
defrosting	cooking	*function*	lion	windows	windows
cook	popcorn	*bag*	driver	xp	xp
cooking	cook	*feature*	xp	driver	*cd*
meat	*sensor*	potato	ubuntu	software	installation
baked	setting	setting	ralink	*website*	*machine*
chicken	adjustment	*kernel*	linux	*hardware*	*disk*
pizza	menu	*number*	realtek	*cd*	*file*
soup	plate	*machine*	mac	*pc*	vista
sens	*inverter*	basic	latest	installation	*disc*
TV (Screen & Image)			Food (Ingredient)		
WE-LDA	AMC	LDA	WE-LDA	AMC	LDA
screen	crisp	screen	olive	flavor	*taste*
bright	picture	*side*	coconut	*sweet*	salt
vivid	sharp	*big*	rosemary	salt	ingredient
sharp	*sound*	*line*	beef	*market*	*natural*
color	*speaker*	*flat*	sauce	fruit	*healthy*
vibrant	clear	hdtv	meat	*natural*	garlic
contrast	image	image	bread	spice	*size*
clear	beautiful	*top*	*cooking*	*strong*	*life*
beautiful	color	lcd	chicken	ingredient	*substitute*
picture	*loud*	resolution	fruit	*delicious*	onion
Music (Feeling)			Clothing (Color)		
WE-LDA	AMC	LDA	WE-LDA	AMC	LDA
voice	compliment	*song*	*hat*	red	color
haunting	gorgeous	*cd*	pink	*side*	*light*
intro	beautiful	*collection*	green	blue	*hat*
yearning	*piece*	wonderful	color	color	black
playing	*person*	*review*	orange	*hat*	*weight*
beautiful	*picture*	worth	*choice*	*open*	red
ella	lovely	lover	bright	*front*	*picture*
dreamy	*page*	*steve*	yellow	*strap*	green
expressive	*wood*	*today*	red	black	blue
gorgeous	*book*	*exception*	*compliment*	green	white

Table 5
Average Topic Coherence values of different schemes for computing word relatedness $r(w_1, w_2)$. Improvements of using both word vector similarity and PMI are all significant ($p < 10^{-4}$) based on 2-tailed paired t-test.

Dataset	Word vector similarity only	PMI only	both
Electronic	−142.63	−136.87	−130.10
Non-electronic	−137.57	−134.60	−129.34

Tables 2 and 3 show the average Topic Coherence of each model using top M words with $M = \{10, 15, 20, 25, 30\}$ respectively. Each value is the average over the 50 domains in the dataset. From Table 2, we can see that WE-LDA performs the best on electronic products dataset and has the highest Topic Coherence values with top 10 words, which shows that WE-LDA finds better quality topics than the baselines. AMC and LTM perform better than LDA but worse than WE-LDA, showing their ability of improving topic interpretability to some extent. WE-LDA outperforms AMC and LTM mainly because its knowledge mining method provides sufficient high quality must-links with word similarities, while AMC and LTM use topical word frequent itemset mining which ignores the word order in text, and could not give adequate must-links with word similarities. Moreover, with word vectors, WE-LDA's must-link clustering is more effective than word frequent itemset mining of AMC and LTM when determining the relatedness of topics and must-links. Probase-LDA performs well because it utilizes a large scale knowledge base, the knowledge is quite sufficient, but the top 10 words' coherence score is lower than WE-LDA. GK-LDA performs slightly better than LDA, indicating its limited capability of handling provided knowledge. LDA-GPU does not perform well because it uses co-document frequency. Since frequent words usually have high co-document frequency with many other words, the frequent words are ranked top in many topics. LF-LDA's performance is not very satisfactory, it uses a switch mechanism to generate words from both regular LDA-style topics and word embeddings, which may assign the same topic to words that are not closely related. Table 3 shows similar results on non-electronic products dataset.

To sum up, we can say that the proposed WE-LDA model can generate higher quality topics than all baseline models. Improvements of WE-LDA on top 10 words are all significant ($p < 10^{-5}$) based on 2-tailed paired t-test.

4.3. Human evaluation

Since our goal is to discover more meaningful topics, here we evaluate the topics based on human judgment. Following Chen and Liu (2014b, a) and Yao et al. (2015), we invited two graduate students who are fluent in English and familiar with Amazon products and reviews to label the generated topics. As we have a lot of domains (100), we selected 10 domains for labeling. The selection was based on the knowledge about the products of the two graduate students. Without adequate knowledge, the labeling will not be reliable. We labeled the topics produced by WE-LDA, AMC and LDA. For labeling, we followed the protocol in Mimno et al. (2011).

Topic Labeling. We first asked the two judges to label each topic as good or bad. Each topic was showed as a list of 10 most probable words. In general, a topic was labeled as good if it had more than half (5) of its words coherently related to each other representing a semantic concept together; otherwise bad.

Word Labeling. The topics labeled as good by both judges were then used for word labeling. Each topical word was labeled as correct if it was coherently related to most words of the topic; otherwise wrong.

The Cohens Kappa agreement score for topic labeling is 0.817, and the Cohens Kappa agreement score for word labeling is 0.806.

Since topics are essentially rankings of words, an intuitive way to evaluate topics is to use $Precision@n$ (or $p@n$), a widely used metric for information retrieval (e.g., web page ranking) and recommendation system. $Precision@n$ is the proportion of correct words in top n words of a topic, we compute the average $Precision@n$ of good topics in each domain. The measure was also used in Chen et al. (2013a), Chen and Liu (2014b, a) and Yao et al. (2015).

Fig. 1 (top) shows that WE-LDA discovers more good topics than AMC and LDA. On average, WE-LDA discovers 1.0 more good topics than AMC and 2.6 more good topics than LDA over the 10 domains. Fig. 1 (middle & bottom) also gives the average $Precision@5$ ($p@5$) and $Precision@10$ ($p@10$) of topical words of only good topics (bad topics are not considered) for each model in each domain. It is obvious that WE-LDA achieves the highest $p@5$ and $p@10$ values for all 10 domains. AMC is also better than LDA in general, but clearly worse than WE-LDA. On average, for $p@5$ and $p@10$, WE-LDA improves AMC by 6.53% and 6.69%, and improves LDA by 13.49% and 11.40% respectively. Significance testing using 2-tailed paired t-tests shows that the improvements of WE-LDA are significant over AMC ($p < 10^{-5}$) and LDA ($p < 10^{-7}$) on $p@5$ and $p@10$. The human evaluation results are consistent with the Topic Coherence results.

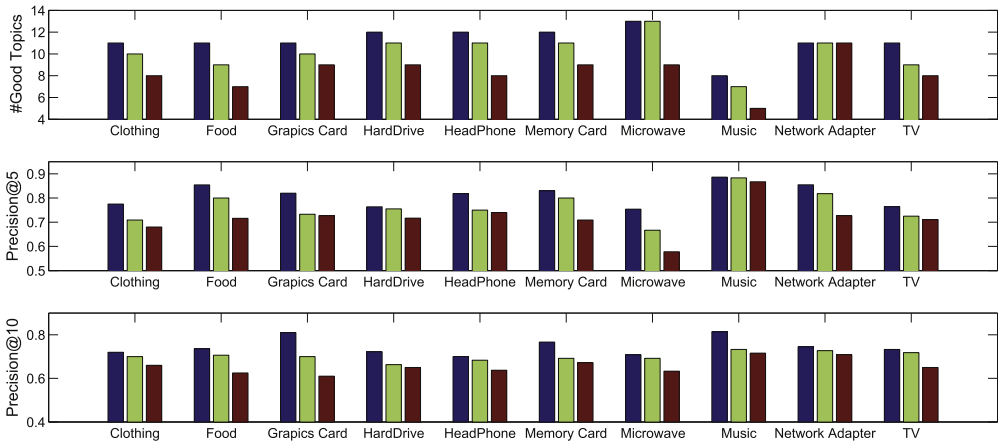


Fig. 1. #Good Topics, Topical words *Precision* @5 and *Precision* @10 of good topics of each model. The bars from left to right in each group are for WE-LDA, AMC and LDA.

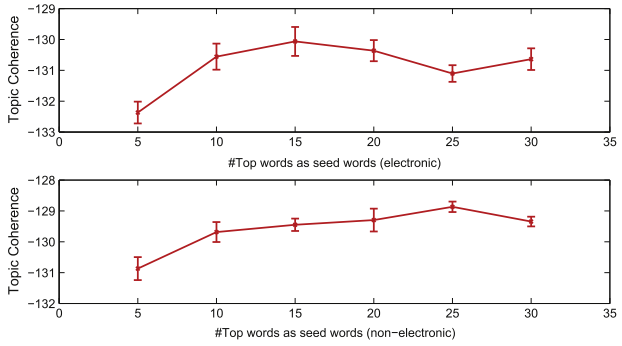


Fig. 2. Average Topic Coherence of top 10 words with different number of seed words on electronic products dataset (top) and non-electronic products dataset (bottom).

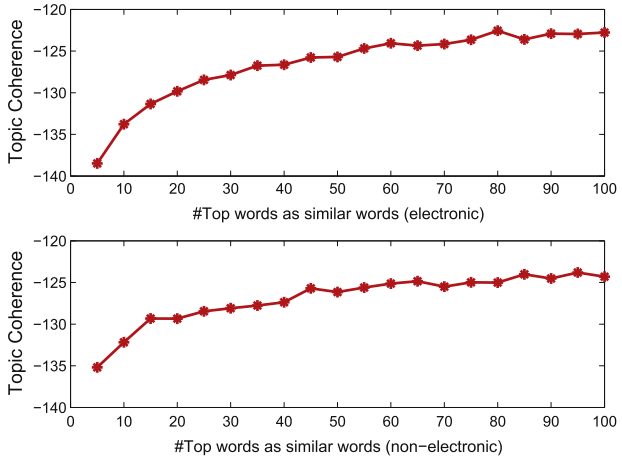


Fig. 3. Average Topic Coherence of top 10 words with different number of similar words on electronic products dataset (top) and non-electronic products dataset (bottom).

4.4. Example topics

This subsection shows some example topics produced by WE-LDA, AMC, and LDA in several domains to give a taste of improvements made by WE-LDA. Each topic is shown with its top 10 words. Errors are italicized and marked in red. From Table 4 ("Microwave (Food Cooking)" means topic "Food Cooking" in domain "Microwave"), we can see that WE-LDA discovers many more meaningful and correct topical words at the top than the baselines. Note that for WE-LDA's topics that were not discovered by the baseline models, we tried to find the best possible

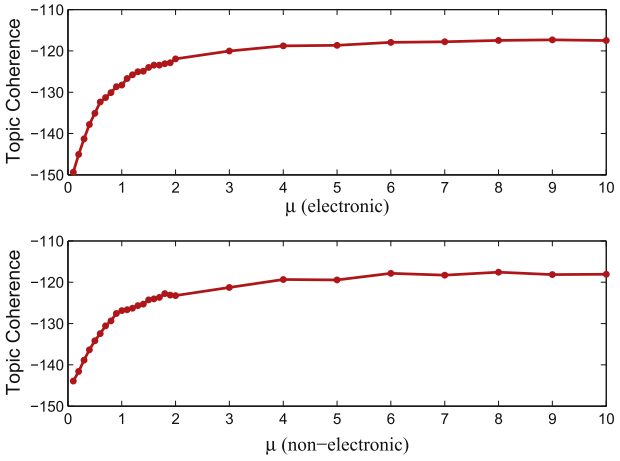


Fig. 4. Average Topic Coherence of top 10 words with different μ value on electronic products dataset (top) and non-electronic products dataset (bottom).

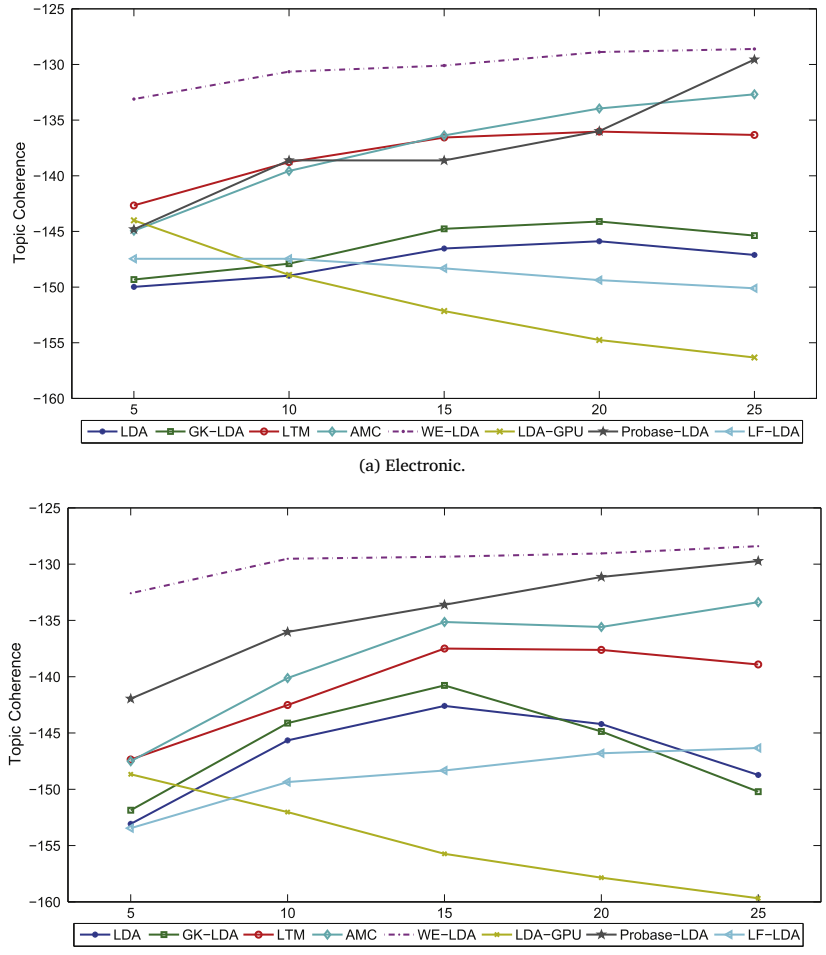
matches from the topics of the baseline models. From the table, we can clearly see that WE-LDA discovers more coherent topics than AMC and LDA. Apart from Table 4, many topics are substantially improved by WE-LDA, including some widely shared topics such as Warranty and Brand.

4.5. Sensitivity to parameters

In this part, we investigate the sensitivity of the three parameters of WE-LDA: the number of top seed words n, the number of most similar words m and the trust score μ. Since the top 10 words under a topic have represented most important semantic information, we only report average Topic Coherence of top 10 words with different parameters, each value is the average over all 50 domains in a dataset. When varying one parameter, we fix all other parameters as in experimental settings.

Fig. 2 shows the average Topic Coherence on the two datasets with $n = \{5, 10, 15, 20, 25, 30\}$ respectively. We can note that using $n = 15$ top words as seed words results in the highest Topic Coherence on electronic products dataset, and using $n = 25$ top words as seed words leads to the highest Topic Coherence on non-electronic products dataset. This means that too few seed words could not generate sufficient knowledge, while too many seed words may generate some incorrect knowledge.

Fig. 3 gives the average Topic Coherence with m from 5 to 100. We observe that with more similar words, Topic Coherence increases rapidly at first, then becomes stable and gets the highest value at $m = 80$ on electronic products dataset. On non-electronic products dataset, the



(a) Electronic.

(b) Non-Electronic.

Fig. 5. Average Topic Coherence of top 10 words with different number of topics on electronic products dataset (top) and non-electronic products dataset (bottom).

curve is similar, and the highest score is at $m = 95$. This demonstrates the must-links produced by a seed word and its most similar words are the primary source of high quality knowledge, while less similar words make little or no contribution to the result. The similarity and PMI score measure the quality of a must-link and guarantee that the WE-LDA model is fed with proper knowledge.

Fig. 4 presents the average Topic Coherence with μ varying from 0.1 to 10. We can see a similar phenomenon as in Fig. 3, i.e., with a higher trust score, Topic Coherence value has a drastic increase at the beginning, then grows slowly and tends to be stable. This shows that the generated must-link knowledge bases are of relatively high correctness and should be trusted by the topic model.

We also explore the effects of word vector similarity and PMI value in word relatedness $r(w_1, w_2)$. Table 5 shows the average Topic Coherence when considering word vector similarity only ($r(w_1, w_2) = sim(w_1, w_2)$ for must-links with both positive PMI values and non-positive PMI values), considering PMI only (using seed words to generate must-links with positive PMI values, and $r(w_1, w_2) = PMI(w_1, w_2)$), and considering both of word vector similarity and PMI values ($r(w_1, w_2) = PMI(w_1, w_2) \times sim(w_1, w_2)$). We can see that considering both of word vector similarity and PMI leads to better results than considering one of them, which shows the correctness of using must-links which carry both general domain knowledge and domain-specific knowledge.

4.6. Effects of number of topics

This subsection compares WE-LDA with the seven state-of-the-art topic models using different number of topics.

The eight models (LDA, GK-LDA, LDA-GPU, LTM, AMC, Probase-LDA, LF-LDA and WE-LDA) are performed with different number of topics. Each model was run with $T \in \{5, 10, 15, 20, 25\}$, others are all the same as in experimental settings.

Fig. 5 shows the average Topic Coherence of each model given different number of topics on two datasets. Each value is the average over all 50 domains in a dataset. We note that given different number of topics, WE-LDA consistently achieves higher Topic Coherence scores than the baseline models on two datasets, which shows the proposed method is robust with different number of must-link clusters. On electronic products dataset, Probase-LDA performs the best at $T = 10, 25$, AMC performs the best at $T = 15, 20$, and LTM performs the best at $T = 5$ among baseline models, but all not as good as WE-LDA. GK-LDA and LF-LDA can perform slightly better than LDA with word embeddings information. LDA-GPU performs well at $T = 5$ when topics are general and topical words are frequent ones, but does not perform well when T increases, and topics tend to be specific. On non-electronic products dataset, the result is similar, Probase-LDA performs the best among baseline models, while it is worse than WE-LDA. Improvements of WE-LDA over Probase-LDA, AMC, LTM and others are all significant ($p < 0.007$) based on 2-tailed paired t-test.

5. Conclusion

This paper has presented WE-LDA, which combines topic model and word embeddings, in particular LDA model and Skip-Gram. The proposed method models document-level word co-occurrence with

knowledge encoded by word vectors automatically learned from a large amount of relevant text data, could extract more coherent topics. Experimental results on real world e-commerce datasets show the effectiveness of the proposed method. We can conclude that semantic similarity of word vectors learned from large-sale data can overcome the sparsity problem of small dataset. We also showed that considering most similar words of a target word can produce coherent topics.

Future work could be exploring more principled and efficient ways to incorporate word embeddings. It is also interesting to experiment with other datasets and word embedding methods rather than e-commerce domain data and Skip-Gram.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61572434), China Knowledge Centre for Engineering Sciences and Technology (No. CKCEST-2017-1-3), Zhejiang Provincial Natural Science Foundation of China (No. LY14F020027) and Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP) (20130101110136).

References

Andrzejewski, D., Zhu, X., 2009. Latent dirichlet allocation with topic-in-set knowledge. In: Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing. Association for Computational Linguistics, pp. 43–48.

Andrzejewski, D., Zhu, X., Craven, M., 2009. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In: ICML. ACM, pp. 25–32.

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: IJCAI, Vol. 22, (1), p. 1171.

Baroni, M., Dinu, G., Kruszewski, G., 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In: ACL, Vol. 1.

Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C., 2003. A neural probabilistic language model. J. Mach. Learn. Res. 3, 1137–1155.

Blei, D.M., Ng, A.Y., Jordan, M.I., 2003. Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022.

Chang, J., Gerrish, S., Wang, C., Boyd-graber, J.L., Blei, D.M., 2009. Reading tea leaves: How humans interpret topic models. In: NIPS, pp. 288–296.

Chemudugunta, C., Holloway, A., Smyth, P., Steyvers, M., 2008. Modeling Documents by Combining Semantic Concepts with Unsupervised Statistical Learning. Springer.

Chen, Z., Liu, B., 2014a. Mining topics in documents: standing on the shoulders of big data. In: KDD. ACM, pp. 1116–1125.

Chen, Z., Liu, B., 2014b. Topic modeling using topics from many domains, lifelong learning and big data. In: ICML, pp. 703–711.

Chen, Z., Mukherjee, A., Liu, B., 2014. Aspect extraction with automated prior knowledge learning. In: ACL, pp. 347–358.

Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R., 2013a. Discovering coherent topics using general knowledge. In: CIKM. ACM, pp. 209–218.

Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R., 2013b. Exploiting domain knowledge in aspect extraction. In: EMNLP, pp. 1655–1667.

Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., Ghosh, R., 2013c. Leveraging multi-domain prior knowledge in topic models. In: IJCAI. AAAI Press, pp. 2071–2077.

Chuang, J., Gupta, S., Manning, C., Heer, J., 2013. Topic model diagnostics: Assessing domain relevance via topical alignment. In: ICML, pp. 612–620.

Collobert, R., Weston, J., 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In: ICML. ACM, pp. 160–167.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural language processing (almost) from scratch. J. Mach. Learn. Res. 12, 2493–2537.

Das, R., Zaheer, M., Dyer, C., 2015. Gaussian LDA for topic models with word embeddings. In: ACL, pp. 795–804.

Fang, A., Macdonald, C., Ounis, I., Habel, P., 2016. Using word embedding to evaluate the coherence of topics from Twitter data. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp. 1057–1060.

Hofmann, T., 1999. Probabilistic latent semantic indexing. In: SIGIR. ACM, pp. 50–57.

Hu, Y., Boyd-Graber, J., Satinoff, B., 2011. Interactive topic modeling. In: ACL.

Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y., 2012. Improving word representations via global context and multiple word prototypes. In: ACL. Association for Computational Linguistics, pp. 873–882.

Mahmoud, H., 2008. Pólya urn Models. CRC Press.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119.

Mimno, D., Wallach, H.M., Talley, E., Leenders, M., McCallum, A., 2011. Optimizing semantic coherence in topic models. In: EMNLP. Association for Computational Linguistics, pp. 262–272.

Mnih, A., Hinton, G., 2007. Three new graphical models for statistical language modelling. In: ICML. ACM, pp. 641–648.

Newman, D., Bonilla, E.V., Buntine, W., 2011. Improving topic coherence with regularized topic models. In: NIPS, pp. 496–504.

Newman, D., Lau, J.H., Grieser, K., Baldwin, T., 2010. Automatic evaluation of topic coherence. In: NAACL-HLT. Association for Computational Linguistics, pp. 100–108.

Nguyen, D.Q., Billingsley, R., Du, L., Johnson, M., 2015. Improving topic models with latent feature word representations. Trans. Assoc. Comput. Linguist. 3, 299–313.

Yang, Y., Downey, D., Boyd-Graber, J.L., Graber, J.B., 2015. Efficient methods for incorporating knowledge into topic models. In: EMNLP, pp. 308–317.

Yao, L., Zhang, Y., Wei, B., Li, L., Wu, F., Zhang, P., Bian, Y., 2016. Concept over Time: the combination of probabilistic topic model with wikipedia knowledge. Expert Syst. Appl. 60, 27–38.

Yao, L., Zhang, Y., Wei, B., Qian, H., Wang, Y., 2015. Incorporating probabilistic knowledge into topic models. In: PAKDD. Springer, pp. 586–597.