Journal of Web Engineering, Vol. 16, No.7&8 (2017) 653-675

DISCOVER SEMANTIC TOPICS IN PATENTS WITHIN A SPECIFIC DOMAIN

WEN MA[1], XIANGFENG LUO[1]*, JUNYU XUAN[2], RUIRONG XUE[1], YIKE GUO[3]

[1]*School of Computer Engineering and Science, Shanghai University, Shanghai, China*
[2]*Faculty of Engineering and Information Technology, University of Technology, Sydney (UTS) Australia*
[3]*Department of Computing, Imperial College London, London, UK*
{mpp1218, luoxf}@shu.edu.cn, Junyu.Xuan@uts.edu.cn, xueruirong@i.shu.edu.cn,
*y.guo@imperial.ac.uk, *CORRESPONDING ADDRESS: luoxf@shu.edu.cn*

Received November 9, 2016
Revised March 30, 2017

Patent topic discovery is critical for innovation-oriented enterprises to hedge the patent application risks and raise the success rate of patent application. Topic models are commonly recognized as an efficient tool for this task by researchers from both academy and industry. However, many existing well-known topic models, e.g., Latent Dirichlet Allocation (LDA), which are particularly designed for the documents represented by word-vectors, exhibit low accuracy and poor interpretability on patent topic discovery task. The reason is that 1) the semantics of documents are still under-explored in a specific domain 2) and the domain background knowledge is not successfully utilized to guide the process of topic discovery. In order to improve the accuracy and the interpretability, we propose a new patent representation and organization with additional inter-word relationships mined from *title*, *abstract*, and *claim* of patents. The representation can endow each patent with more semantics than word-vector. Meanwhile, we build a Backbone Association Link Network (Backbone ALN) to incorporate domain background semantics to further enhance the semantics of patents. With new semantic-rich patent representations, we propose a Semantic LDA model to discover semantic topics from patents within a specific domain. It can discover semantic topics with association relations between words rather than a single word vector. At last, accuracy and interpretability of the proposed model are verified on real-world patents datasets from the United States Patent and Trademark Office. The experimental results show that Semantic LDA model yields better performance than other conventional models (e.g., LDA). Furthermore, our proposed model can be easily generalized to other related text mining corpus.

Key words: Patent topic discovery, Latent Dirichlet Allocation, Backbone Association Link Network, Domain knowledge

Analysis *Communicated by*: M. Gaedke & Q. Li

1 Introduction

Patent topic discovery is a key issue in patent knowledge mining as it is beneficial to innovation-oriented enterprises, decision makers, and so on [1]. Patent topic analysis can not only identify novel patents [2] and analyse technology distribution [3], but also track and predict technology evolution process [4][5]. For example, in *refrigerator* domain, there are numerous critical technologies, such as

energy-saving, *fresh-keeping*, *intelligence* etc. Through mining the patent knowledge in *refrigerator* domain, we could discover a number of hidden topics that can further contribute to identify critical technologies and improve enterprise invention ability significantly. In a specific domain, discovering key techniques and their relationships by topics can help enterprises enhance product innovation and identify potential competitors. Therefore, topic-level patent analysis in a specific domain becomes more and more important on quickly and accurately discovering vital technologies [6].

Patent documents have been provided basic category information with predefined taxonomy code for efficient patent analysis [7], e.g., CPC (Cooperative Patent Classification). Taxonomy code can help to fast and accurately retrieve patent document. However, these taxonomy codes are too rigid and general as patent topics. For instance, in *refrigerator* domain, a patent with multiple classification can be in the field of "*A*" and "*G*" ("*A*" represents human necessities and "*G*" represents physics) etc. However, it is inefficient to analyze this domain from the patents in "*A*" and "*G*", because most of patents in "*A*" and "*G*" are irrelevant with the *refrigerator*. For the emerging topic of technology, the taxonomy codes of patents are unknown to researchers. It is impossible to automatically find relevant patents from massive documents. Therefore, the topics of technologies are still difficult to be analysed by taxonomy code.

Table 1: The topic discovered by traditional LDA shows as a bag of simplicity words with weak semantics which cannot be used to better explain topics.

word	*probability*
graphene	*0.0246*
hydrazine	*0.0212*
emit	*0.0211*
cyclohexane	*0.0113*
semimetal	*0.0018*
fluoroplastic	*0.0015*
detector	*0.0013*
memristor	*0.0011*

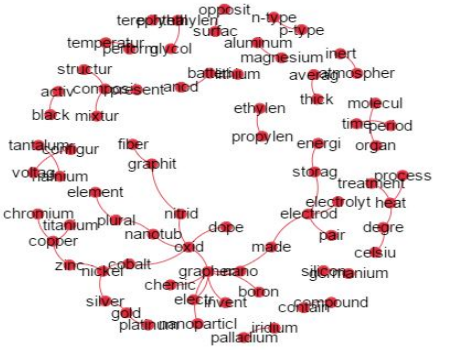


Fig.1 The topic discovered by the proposed Semantic LDA model has rich semantics, which shows as a network consisted of detailed keywords and association relationships.

Existing patent topic discovery models could be mainly categorised as follow: 1) distance-based models, such as K-means [8]; 2) density-based models, such as density-based spatial clustering of applications with noise [9]; 3) hierarchical agglomerative clustering [10]; and 4) probabilistic models, e.g., latent dirichlet allocation [11] and probabilistic latent semantic analysis [12]. Besides, a number of patent retrieval systems, such as Google Patent and PatentMiner [13], have provided an efficient search for patents based on discovered topics.

However, the above mentioned models often mingle several different concepts/domains to discover patent topics. The performance often gets worse when the patents are limited in a specific domain, because domain words are widely distributed throughout all patent documents which make the semantic distinction ability of these words much weaker than in multi-domains. Hence, it is an enormous challenge to discover latent topic in a specific domain. What's more, these models, e.g., LDA, usually highlight the word frequency. It is obviously inadequate that they represent patent knowledge just using bag-of-words as the only feature. Besides, topics discovered by bag-of-words based models suffer from problem of poor interpretability. As shown in Table 1, the topic that is represented by a word vector which has poor interpretability/semantics, because it is hard to reveal the knowledge association and help researchers dig implicit knowledge in a specific domain.

In this paper, to overcome the above limitations of traditional models, we propose an innovative probabilistic topic model: Semantic LDA. Based on classical LDA, our model involves Association

Link Network (ALN) [14] to discover patent semantic topics. By incorporating word relationships, ALN can provide rich patent semantics. ALN's relationships are derived to discover semantic topics, and can express more semantics than only use bag-of-words. Hence, we name the model as Semantic LDA. Specifically, the discovered topics are determined by not only the bag-of-words but also the extracted word relationships, which can be called *semantic topics*. As shown in Fig.1, Semantic LDA can generate semantic topics involving words and words' association relationships. In particular, relationships and weights between words can be clearly revealed.

The main contributions of this paper are:

1) Patent semantic topics discovered by the proposed Semantic LDA model have better interpretability and rich semantics, which can reveal the explicit and implicit knowledge of the patent documents in a specific domain.
2) Domain background knowledge mined and represented by Association Link Network is utilized to improve the accuracy and the interpretation of topic discovery process.

The rest of this paper is structured as follows: Sections 2 reviews the related works. In section 3, we introduce the overview of our model and its Gibbs sampling-based inference algorithm. The experiment setup and results are summarized in Section 4, and we also include a case study to show the semantic topic result in this section. Finally, we draw our conclusions and present possible future studies in Section 5.

2 Related Work

In this section, we introduce the structure of patents in section 2.1, and the existing well-known models for patent topic discovery in section 2.2. Section 2.3 is mainly about Association Link Network (ALN) based models.

2.1 Patent Structure in Technological Research

A typical patent document often contains several necessary sections, including title, front page (announcement, bibliography, classification, abstract, etc.), detailed specifications, claims, declaration, and/or a list of drawings to illustrate the idea of the solution. The patent document is often lengthy, taking advantage of all sections to mine the patent knowledge is inefficiency and noisy. In this work, we take three sections of the patent to discover meaningful semantic topics. Because these sections contain more abundant information than the other sections, e.g., 1) *title* includes a few core words, can be regarded as the patent keywords in a paper, 2) *abstract* contains the patent core concept and describes the invention methods in brief paragraph, 3) *claims* as the most significant part of the semantic distinction, declare the listed items in this patent which are protected by law. This paper utilizes patent *title*, *abstract* and *claims* as the main resources for topic discovery.

2.2 Patent Topic Discovery Models

Topic discovery has been studied for years. Existing researches on patent topic discovery can be divided into 1) probabilistic models and 2) non-probabilistic models.

2.2.1 Probabilistic Models

In probabilistic models, the document is represented as the bag-of-words because of its simplicity. Even the outstanding probabilistic models, such as probabilistic latent semantic analysis (PLSA) and latent Dirichlet allocation (LDA) are no exception. In the patent area, a variety of LDA-based models depending on the different aspects of patent data are widely adopted. Inventor-Company-Topic (ICT) Model [15] combined the patent features with LDA-based topic model to discover latent semantic topics. Subhashini et al. [16] used the topic model to offer a reduced-form representation of the knowledge content in a patent. Kim et al. [17] applied LDA to analyse technological trend. Segmented

topic model (STM) [18] took advantage of documents' structure to explore patents correlated segment topics. Xuan et al. [19] used Bernoulli distributions to model the edges between nodes in a graph, it can describe graphs better than the ones from LDA.

2.2.2 Non-probabilistic Models

In non-probabilistic models, Kim et al. [20] clustered patent documents keywords by k-Means, and then built a semantic network of keywords for patent analysis. Che et al. [21] used neural network based approach for discovering patent topics, however, the approach is less precise when data is large. Shih et al. [22] constructed the patent ontology network by calculating four types of nodes and eight types of edges relationships, and discovered topics by extracting k-nearest neighbour. Chen et al. [23] proposed a fuzzy set based topic development measurement (FTDM) model to estimate and evaluate the topics.

To sum up, although various models have considered several characters of patent documents, they do not perform well on patent topic discovery for the following reasons: 1) traditional models pay less attention to investigate the effect of patent content semantics between words association relationships; 2) topic discovery process has not included the domain patent knowledge successfully. In order to address these hiatus, this paper emphasizes on the words association relationships, and utilizes the domain knowledge as background knowledge to discover patent topic in a specific domain.

2.3 ALN-based Models

ALN is a type of semantic link network to organize various resources, which can briefly represent the knowledge of documents. It is combined by semantic nodes and semantic chain which links the node. The node can be a document, a web page or even a website. ALN is composed of associated links between nodes, and it can be represented as follows:

$$ALN = < N, L > \quad (1)$$

where N is a set of the resource nodes, and L is a set of weighted semantic links. The keyword level ALN is a weighted network, in which a node represents a word and the edges represent word relationships. In the network, nodes semantically related always links with each other, the links strength represents the relevancy between them. ALN-cedm [24] is one of the well-known ALN-based models, which builds keyword-level ALN to represent the core semantics, the method can be used to timely discover newly occurring hot events.

In ALN, for example, when <*electronic, device*> are discovered as strongly correlated relationships, either as <*semiconductor, device*>, it is highly probable that these relationships are related to one topic, e.g., "*device*". Therefore, we can use keyword-level ALN to help discover topics with more accurate and semantic.

3 Semantic LDA Model

In this section, we first present the overall framework for discovering patent semantic topics, and each detailed step is illustrated. Then we introduce Semantic LDA model in detail, and followed by Gibbs sampling for this model.

3.1 Framework

The overall framework of discovering semantic topic in patent documents is shown in Fig.2. Firstly, we acquire the target words (e.g. *graphene*) of a specific domain, and collect relative patents. Then we extract the patents' *title*, *abstract*, *claim* and CPC code as the resource for topic discovery. As the words and relationships can provide more rich semantics of a patent, we present the patent knowledge with words and relationships. At stage 2, we extract the patent keywords by considering domain characteristic. Stage 3 builds backbone association link network by combining domain association link

network with patent association link network. The detailed procedures can be seen in Section 3.2.2. Subsequently, by incorporating patent keywords and relationships from backbone association link network, we construct a Semantic LDA model to discover topics of patents. Finally, we use the Gibbs sampling to train this model to estimate the unknown parameters.

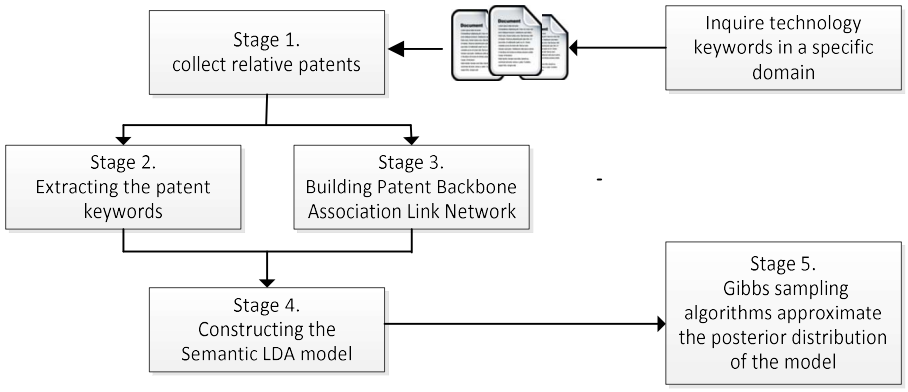


Fig.2 Framework architecture of discovering patent semantic topics in a specific domain

3.2 Patent Feature Selection

As we discussed previously, we consider that a patent knowledge can be represented by words and their association relationships. In various domains, the features of the patent knowledge are diverse. In this paper, we extract 1) patent keywords and 2) patent word relationships based on domain features to represent a patent knowledge.

3.2.1 Patent Keywords

Traditional models pay little attention to words' feature based on various domains, except the word frequency. Actually, in a specific domain, each word's semantic discrimination capability is diverse. We have known that the document frequency distribution of words follows the power-law distribution [25]. As shown in Fig.3, in a domain, the words can be divided into three parts: 1) high frequency words which occur many times in nearly each patent of this domain; 2) medium frequency keywords which can represent the main semantics of the patent; 3) low frequency words ,which always change over time, reveal the new knowledge and technologies. Generally, high frequency words are too common to be used in distinguishing semantics between different topics in a specific domain. Herein, high frequency words do not mean common stop words. High frequency words are determined by the domains, but make little contribution to the topic discovery. To some extent, using these high frequency words will decrease the accuracy of discovering topics.

Therefore, the medium and low frequency words are deemed to be better candidates for topic discovery. This paper uses the two modules to select patents' representative keywords in our model:

1) Remove stop words such as *digital number*, *that*, *these*, *or*, *and*, etc.
2) Remove comparatively high frequency words that are lack of semantic distinction in a specific domain.

For example, Fig.4 lists the top 17 stemming keywords of a graphene domain. It is obvious that silicon, graphene, oxide, graphite, etc. occurs in most of the patents. The topic results mostly contain these words which make little contribution to discover topic.

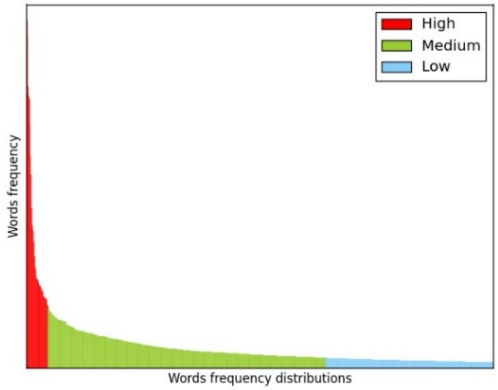


Fig.3 Words document frequency distributions in a specific domain, the high frequency words occurs nearly each patent document with weak distinguish ability.

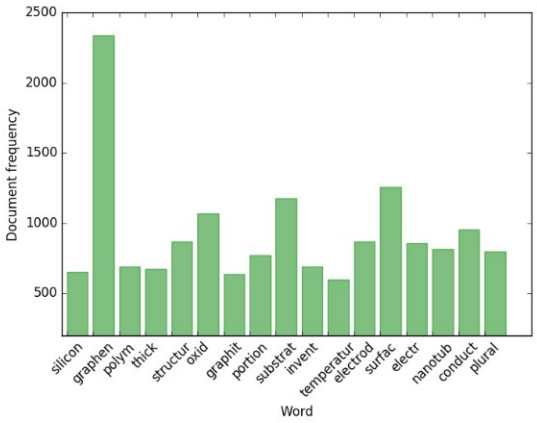


Fig.4 Top 17 words distinction with weak semantics in the *graphene* domain, since these words are too common at each patent to discover topics for their poor discrimination capability.

3.2.2 Patent Word Relationships

Bag-of-words based topic models can only get the distribution of words which usually lack of semantics. To overcome the limitation, in this paper, we add the word association relationships into the topic model. It takes three steps to mine a patent's association relationships between words 1) building Domain Association Link Network (Domain ALN) to represent the domain background knowledge; 2) building Patent Association Link Network (Patent ALN) to mine internal semantics of a patent; 3) building Backbone Association Link Network (Backbone ALN) by combining Domain ALN with Patent ALN to enrich a patent semantics in a specific domain. Finally, besides the patent words extracted from section 3.2.1, the patent's Backbone ALN will be incorporated into to form the Semantic LDA model.

3.2.2.1 Building Domain Association Link Network (Domain ALN)

Domain background knowledge contains high level semantics in a specific domain, it reveals the implicate relations between patents. In order to find implicate knowledge of the domain, we build the Domain ALN. However, it is inefficient and noisy to use all parts of patent structure to build the

Domain ALN. For a patent, because the *title* information is the essence of the whole patent, the collection of a patent *title* can be best candidate to mine the core semantics of this domain.

Definition 1 *Domain Association Link Network (Domain ALN)*: A specific domain association link network is defined as an undirected graph $\mathcal{G} = (\mathcal{W}, \mathcal{E})$ which is composed of a pair of sets, where $\mathcal{W}$ is the set of words all coming from the fixed vocabulary $\mathcal{V}$. $\mathcal{E}$ is the edge between words, where $(\mu, \nu) \in \mathcal{E}, \mu \in \mathcal{W}, \nu \in \mathcal{W}$, the edge indicates the occurrence of semantic interactions between the corresponding word terms, the value describes the strength of such interaction.

The association strength between keywords is defined as:

$$e^{\mathcal{C}}_{w_i - w_j} = \frac{Co(w_i, w_j)}{\sqrt{DF(w_i) DF(w_j)}} \tag{2}$$

where $\mathcal{C}$ is the *title* set of the patents, $Co(w_i, w_j)$ is the total co-occurrence frequency of word w_i and w_j in $\mathcal{C}$. $DF(w_i)$ is the word w_i occurrence frequency in the $\mathcal{C}$. As Fig.5 shows, we can build the Domain ALN as follows:

Algorithm 1: *Building Domain ALN of a specific domain*

Input: $c_i | i = 1, \ldots, n$

Output: Domain ALN$|(\mu, \nu) \in \mathcal{E}, \mu \in \mathcal{W}, \nu \in \mathcal{W}$

1) extract word set $\mathcal{W}$ *from* $\mathcal{C}$*;*

2) for each pair of word $< w_i, w_j >$*:*

3) calculate the $e^{\mathcal{C}}_{w_i - w_j}$ *through Eq. (2);*

4) construct Domain ALN by combine $\mathcal{E}$ *with* $e^{\mathcal{C}}_{w_i - w_j}$*;*

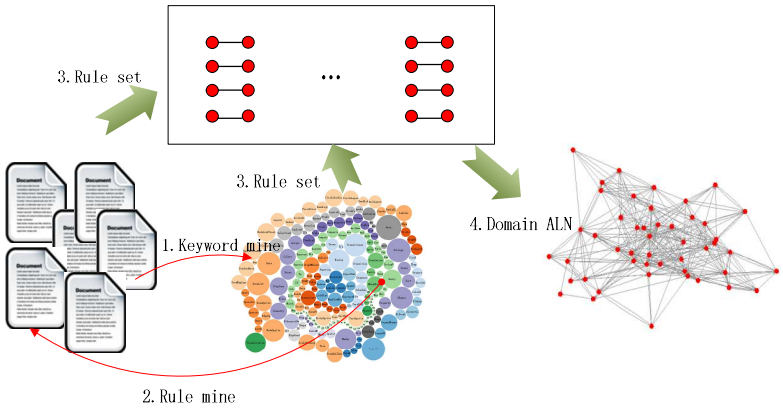


Fig.5 Building a Domain Association Link Network using patent *title* set since it represents the core semantics of this domain, and process is efficient because *title* is the most refined information in a patent

3.2.2.2 Build Patent Association Link Network (Patent ALN)

Abundance patent knowledge is implied in words and relationships. A Patent ALN which consists of keywords and relationships between inter-words of a patent can represent the patent internal semantics. The *title*, *abstract*, *claims* of the patent can represent its summarized information. Based on valuable information, we build Patent ALN to mine the patent knowledge. Through using slide window to

generate transactions [26] of a patent, we could obtain the association rules and build Patent ALN as Fig.6.

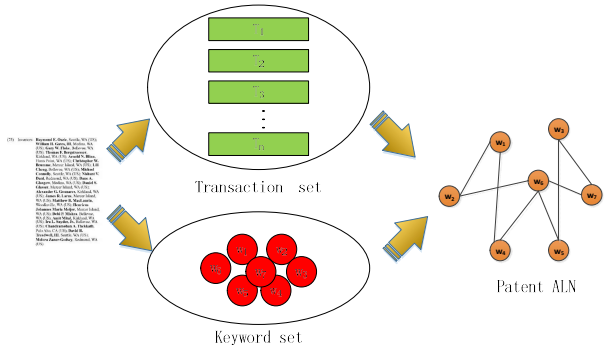


Fig.6 Building Patent Association Link Network by keywords and association rules mined from patent content in a specific domain

Definition 2 *Patent Association Link Network* (*Patent ALN*): Like *ALN*, *Patent ALN i*s graph of the patent words association strength $\mathcal{G}' = (\mathcal{W}', \mathcal{E}')$, where $\mathcal{W}'$ is the set of words, $\mathcal{E}'$ is the edge between words of the patent. A Patent ALN can represent patent words association relations, and describes the strength of such interaction in a patent itself.

The association strength between keywords in the document can be defined as follows:

$$e^{T_n}_{w_i - w_j} = \frac{Co(w_i, w_j)}{\sqrt{DF(w_i) DF(w_j)}} \qquad (3)$$

where T_n is the n^{th} transaction of patent document, $Co(w_i, w_j)$ is the total co-occurrence times that word w_i and word w_j in the transaction of the document. $DF(w_i)$ is the word w_i occurrence times in the T_n。 As Fig.4 shows, we can mine the patent content to build Patent ALN.

***Algorithm 2**: Building a Patent ALN*

Input: patent d_n's title, abstract and claims of a specific domain
Output: a Patent ALN

1)segment d_n by the slide window to generate transaction set T;
2)extract word set W from d_n;
3)for each pair word $< w_i, w_j >$:
4) calculate the $e^{c}_{w_i - w_j}$ through Eq. (3);
5)construct Patent ALN by combine $\mathcal{G}'$ with $e^{T}_{w_i - w_j}$;

3.2.2.3 Build Patent Backbone Association Link Network (Backbone ALN)

A Backbone ALN can represent a patent explicit and implicit knowledge/semantics with background knowledge in a specific domain. We build patent Backbone ALN by combining Domain ALN with Patent ALN with the purpose of balancing the power of both global semantics and patent internal semantics. The process can be described as follows:

***Algorithm 3**: Building patent d_n's Backbone ALN*

Input: d_n's Patent ALN, Domain ALN, $\mathbf{w}$, parameters μ, γ_1 and γ_2
Output: patent d_n's Backbone ALN in a specific domain

1) *extract candidate word set $w_{candidate1}$ from $\mathbf{w}$ which are included in top μ percentage of all the words in d_n;*
2) *extract candidate word set $w_{candidate2}$ from Patent ALN which edges are included in top γ_2 percentage of all relationships in Patent ALN*
3) $w_{candidate} = w_{candidate1} \cap w_{candidate2}$
4) *extract the relationships set e^C by $w_i \in w_{candidate}$ which edges exists in top γ_1 percentage of Domain ALN*
5) *extract the relationships set e^D included in top γ_2 percentage of Patent ALN*
6) *construct Backbone ALN by combine e^C with e^D;*

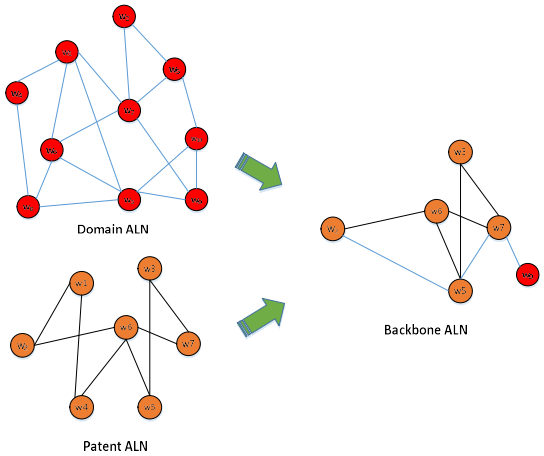


Fig.7 Selecting words in top percentage word set as candidates to enrich semantics in Domain ALN, then combines with Patent ALN to build Backbone ALN of patent in a specific domain

Fig.8 A Backbone ALN which constructed by Domain ALN and Patent ALN

In Algorithm 3, γ_1, γ_2 are the percentage parameters to balance the importance of Domain ALN and Patent ALN. As illustrated in Fig. 7, there is no relation e_{79} ($e_{79} =< w_7, w_9 >$) in Patent ALN, but the word w_7 is selected as the candidate to expand relationship from Domain ALN. As to d_i's Patent ALN, the relationship of e_{14} ($e_{14} =< w_1, w_4 >$) is weaker than threshold which is excluded from top γ_2 percentage. It means that this relationship contributes less importance to this patent, so we cut it down to cohere semantics. Finally, we combine expanded relation from Domain core relations with Paten ALN to represent a patent semantics network. Fig.8 shows a Backbone ALN of patent textual "Method for producing graphene in a magnetic field" in the *graphene* domain.

3.3 Semantic LDA Model

This paper assumes that not only the words but also the word relationships in patents can refine the semantics topics of a domain. In a specific domain, if $\mathcal{D}$ is a patent collection and K is the number of the patent, the words' topic in $\mathcal{D}$ will be defined as $\Phi = \{\phi_k\}_{1 \le k \le K}$, and the words' relation topic in $\mathcal{D}$ will be defined as $H = \{\eta_k\}_{1 \le k \le K}$.

Some other important notations are listed in Table 2.

Table 2: Notations of the patent semantic topic discovery model

Notation	*Description*
w_{d_i}	*the i[th] word in patent d*
r_{d_i}	*the i[th] relationship in patent d*
θ_d	*topic distribution of document in a specific domain*
z_{d_i}	*topic of word i in document d*
ρ_{d_j}	*topic of edge j in document d*
$\emptyset_k$	*word distribution of topic k*
H_k	*edge distribution of topic k*
α, β, γ	*dirichlet prior hyper-parameter*

The Semantic LDA model is also a generative model, the generative process of a patent in a specific domain is as follows:

1) For each patent d_n of a specific domain, n=1,…,N,
   Draw a patent's topic $\theta_d \sim Dirichlet(\alpha)$.
2) Words generative process:
   For each topic t =1,…, K, draw the topic-word distribution $\emptyset_t \sim Dirichlet(\beta)$.
   a) Draw a topic $z_{d_i} \sim Multi(\theta_d)$.
   b) Draw a word $w_{d_i} \sim Multi(z_{d_i})$.
3) Edges generative process:
   For each edge topic q =1,…, K, draw the topic-relation distribution $\eta_q \sim Dirichlet(\gamma)$.
   a) Draw a topic $\rho_{d_i} \sim Multi(\theta_d)$.
   b) Draw a relationship $r_{d_i} \sim Multi(\rho_{d_i})$.

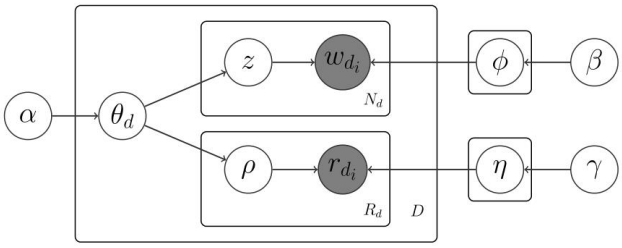


Fig.9 Graphical model of Semantic LDA: w_{d_i} is the patent words extracted by specific domain characteristic, r_{d_i} is the Backbone ALN relations constructed in this domain, unlike traditional LDA, the topic θ is influence by word matrix w and relationship matrix r.

We can see from Fig.9, d_i is one of the patents of $\mathcal{D}$ in a specific domain, w_{d_i} is d_i's keyword with better semantic discrimination capability, r_{d_j} is edge extracted from d_i's Backbone ALN, which constructed by combining Domain ALN with d_i's Patent ALN. The edge definition r_{d_j} is a description of the observed data, so as the keyword w_{d_i}. Different from LDA, our topics are determined not only by the words but also by the extracted edge relations. Semantic LDA model overcomes the drawback of bag-of-words based models by drawing edge topic assignment ρ_{d_j}. Especially, Semantic LDA model considers the domain background knowledge to guide the process of topic discovery. In the next section, the detailed learning process for Semantic LDA is illustrated.

3.4 Gibbs Sampling for Semantic LDA Model

For Semantic LDA model learning, a diversity of methods [27] can be used to estimate the unknown parameters. This section mainly illustrates the detailed process for Gibbs sampling by jointly considering the words and relationships in Semantic LDA model. In the previous section, we described the generative process of the word and edge relationships. For estimating the latent variables, by given observable keywords (***w***) and relationships (***r***) of a patent in the domain, we need to find the conditional distributions for each latent variable in posterior distribution.

In Semantic LDA model, θ_d represents the patent topic distribution of the patent d, it is a probability of each topics, where $\sum_{k=1}^{K}\theta_{dk}=1$. θ_d is a Dirichlet distribution, its posterior conditional distribution on other variables can be observed as,

$$P(\theta_d|z_d,\rho_d,\alpha)\propto\prod_{k=1}^{K}\theta_{dk}^{\sum_{i=1}^{N_d}I_{z_{d_i}=k}+\sum_{j=1}^{R_d}I_{\rho_{d_j}=k}+\alpha_k-1} \tag{4}$$

where z_i represents the topic assignment of w_i, as well as ρ_i represents r_i topic assignment. For all patent documents, $\sum_{i=1}^{N_d}I_{z_{d_i}=k}$ is the number of tokens with word w_i that are assigned to topic k, and $\sum_{i=1}^{N_d}I_{z_{d_i}=k}$ is the number of tokens with relationship r_i that are assigned to topic k.

For the unknown variable φ_k, it is topic k's word vocabulary distribution, the posterior distribution should be,

$$P(\varphi_k|z,w,\beta)\propto\prod_{v=1}^{V}\varphi_{kv}^{\sum_{d=1}^{D}\sum_{i:z_{d_i}=k}I(w_{d_i})+\beta_v-1} \tag{5}$$

where $\sum_{i:z_{d_i}=k}I(w_{d_i}=v)$ is the word item v number under topic k.

As well, we define word relationships having the relationship vocabulary distribution. The posterior distribution can be refined as,

$$P(\eta_k|\rho,r,\gamma)\propto\prod_{d=1}^{D}\prod_{j:\rho_{d_j}=k}\eta_{kv}^{I_{d_j=s}}\prod_{s=1}^{E}\eta_{ks}^{\gamma_s-1} \tag{6}$$

where $\sum_{j:\rho_{d_j}=k} I\left(r_{d_j}=v^t\right)$ is the number of the item v^t which assigned to topic k.

Here, we obtain θ_d, φ_k, η_k, then we sample the word and relation topic z_{d_i}, ρ_{d_j},

$$P\left(z_{d_i}=k\middle|\theta_d, w_{d_i}, \varphi\right)=\theta_{dk}\varphi_{k,w_{d_i}} \quad (7)$$

$$P\left(\rho_{d_j}=k\middle|r_{d_j}, \theta_d, \eta\right)=\theta_{dk}\eta_{k,r_{d_j}} \quad (8)$$

Then we iteratively sample θ_d, φ_k, η_k, z and ρ. The sampling process is summarized in Algorithm 4:

***Algorithm 4**: Gibbs sampling for Semantic LDA model*

Input: The number of topic K, hyper parameters α, β, γ
Output: multinomial distribution φ, η and θ
initialize topic assignments randomly for all the keywords and word relationships
for iter = 1 to N_{iter} do
  if iter>burn-in
   for each document $d \in D$,d = 1…N do:
    draw φ_k from $P(\varphi_k|z,w,\beta)$
    draw η_k from $P(\eta_k|\rho,r,\gamma)$
    draw θ_d from $P(\theta_d|z_d,\rho_d,\alpha)$
    update z in Eq. (7) and ρ in Eq. (8)
compute the distribution φ, η and θ on the sample average

4 Experiments and Results Analysis

We used LDA and ALN-cedm model as the baseline models. In following sections, we conduct experiments on real-world datasets to evaluate the proposed method by exploiting topic discovery results. The detailed datasets are introduced in Section 4.1, whereas we discuss experiment results and detailed analysis in Sections 4.2, 4.3, and 4.4.

4.1. Data Sets

To evaluate the quality of the proposed model, we need public patent datasets in some specific domain. However, such ideal datasets do not exist. Therefore, we programmed the spider to fetch patent documents from U.S. Patent and Trademark Office (USPTO: www.uspto.gov) database, and then manually refined four datasets as follows:

- **Microelectronic Material Dataset** covers patents about "*microelectronic material*" ("*graphene*", "*silicon*"). "*graphene*" and "*silicon*" are two mainly materials in microelectronics industry. Overall, 6206 patents have been collected from the database. Then, four human annotators have manually confirmed these patent documents in the "*microelectronic material*" field and are associated with the appropriate topic, e.g., *graphene* or *silicon*. As showing in Table 3, the result dataset contains 1033 *silicon* patent documents and 1183 *graphene* patent documents.

Table 3: Statistics of patent number in Microelectronic Material dataset

Domain	*Topic name*	*Patent number*	*Label type*
microelectronic material	*graphene*	*1183*	*Single-label*
	silicon	*1033*	*Single-label*

The other three datasets collected by searching keywords "*graphene*", "*car*" and "*bacterial*", which are widely applied in each aspect and belong to three domains. It is difficult to obtain the ground truth for evaluating our model's performance. The CPC classification system is a five-level classification schema. A patent document can cover 9 sections at first level. We extract first level sections of CPC, and map them to multi-labels which have nine dimensions. Table 4 is the patent number and the label type of the datasets. Table 5 shows the sample of the CPC code of the patents.

Table 4: Statistics of patent number in three specific domains

Specific Domain	*Patent Number*	*Label Type*
graphene	*2340*	*Multi-label*
car	*1008*	*Multi-label*
bacterial	*1121*	*Multi-label*

Table 5: The example of patent multi-label, we extract CPC code's first level section as each patent's multi-label, such as "H01M" can be extracted to "H"

Patent Number	*Cooperative Patent Classification*	*Multi-Label (A B C D E F G H Y)*
5,543,021	***H**01M*	*0 0 0 0 0 0 0 1 0*
6,400,091	***B**82Y, **H**01J, **Y**10S*	*0 1 0 0 0 0 0 1 1*
7,014,829	***B**82Y, **D**01F, **Y**10S, **Y**10T*	*0 1 0 0 1 0 0 0 1*

4.2. Experimental on Microelectronic Material Dataset

To verify the effectiveness of Semantic LDA model on single-label topics, we will conduct experiments on Microelectronic Material dataset. Then we will compare our model with LDA.

4.2.1 Evaluation Metrics

In the Microelectronic Material dataset, each patent is a single-labelled document that belongs to one topic. In this experiment, the solution calculates the most possible label. We analyse the results from each of the perspectives of precision, recall, and F-measure [28]. Precision is the fraction of detections which are true positives. Recall is the fraction of true positives which are detected. F-measure is the ultimate measure of performance of the method.

4.2.2 Experimental Results

In this collection, we set the number of topics K=2 for all the models. In the experiment, both in LDA and Semantic LDA model, we removed the words that frequency is one. The recommended value of α in LDA is 50 divided by the number of topics [29], β = 0.1 and γ = 0.1, in our model, special parameters μ= 0.0016, γ_1=0.001 and γ_2= 0.0005. In our Gibbs sampling, we set the number of iterations = 4000, burn-in = 1000 and sample-lag = 4.

Table 6: Average score of three metrics repeated 10 times by Semantic LDA model and LDA, the best performance is printed in bold for each performance measure.

Method	*Precision*	*Recall*	*F1-Measure*
Semantic LDA	***0.7834***	***0.7505***	***0.7665***
LDA	*0.5670*	*0.5473*	*0.5669*

Table 6 presents the results for those models on the Microelectronic Material dataset, in specially, we can observe precision of Semantic LDA is 0.7834. The performance of LDA turns out to be much worse than Semantic LDA model. Evidently, the result illustrates that our domain relationship can highly enhance the topic discovery result.

4.3. Experimental on Graphene Dataset

As shown in the above section, Semantic LDA model can significantly improve the performance of discovering topics in wider domain. For proving that our model can discover better topics at different levels, hence, we will apply our model into a narrower domain dataset.

4.3.1 Evaluation Metrics

It is obviously that the patents in a same topic are more likely assigned to the same CPC code. For a relatively fair comparison, we can judge the estimation from how it fitting the actual CPC code distribution. Considering each patent has multi-label of CPC code, we extract patent's label set $\mathcal{L} = \{(x_i, Y_i) | 1 < i < d\}$ from the patent first level section as, x_i represents a patent document, Y_i is the x_i's label set; $f(x_i, y)$ returns the real-value indicating the confidence for y to be a proper label if x_i; $rank(x_i, y)$ represents the rank of y derived from (x_i, y). Four evaluation metrics widely-used in multi-label learning [30] are employed in this paper:

1) Hamming loss:

$$\mathcal{H}loss_{\mathcal{L}} = \frac{1}{d}\sum_{i=1}^{d}\frac{|Y_i \Delta Y_i'|}{Q} \tag{9}$$

where Y_i is the ground truth, Y_i' is the predicated label set and Δ stands for the symmetric difference between two labels. The hamming loss represent the ratios of patent topic labels are missed or irrelevant labels are predicted.

2) Average precision:

$$ave\_pre_{\mathcal{L}} = \frac{1}{d}\sum_{i=1}^{d}\frac{1}{|Y_i|}\sum_{l_k \in Y_i}\frac{|\mathcal{R}(x_i, l_k)|}{rank(x_i, l_k)}, \text{ where}$$

$$\mathcal{R}(x_i, l_k) = \{l_j | rank(x_i, l_j) \leq rank(x_i, l_k), l_j \in Y_i\} \tag{10}$$

The average precision evaluates the average fraction of proper labels ranked above a particular label $y \in Y_i$.

3) Coverage:

$$Coverage_{\mathcal{L}} = \frac{1}{d}\sum_{i=1}^{d}\max_{l_k \in Y_i} rank(x_i, y_k) - 1 \tag{11}$$

The coverage evaluates how many steps are needed to move down the label list in order to cover all the proper labels of the dataset. It can reveal the step numbers which are need to move down to cover all test dataset relevant labels.

4) Ranking loss:

$$\mathcal{R}loss_{\mathcal{L}} = \frac{1}{d}\sum_{i=1}^{d}\frac{1}{|Y_i||\overline{Y}_i|} \cdot |\mathcal{R}_i| \tag{12}$$

where $\mathcal{R}_i = \{(y_1, y_2) | f(x_i, y_1) \leq f(x_i, y_2), (y_1, y_2) \in Y_i \times \overline{Y}_i\}$

Here $\overline{Y}_i$ represents the complementary set Y_i in $\mathcal{Y}$. Smaller values of $\mathcal{R}loss_{\mathcal{L}}$ metrics correspond to higher classification quality. The ranking loss evaluates the average fraction of label pairs that are disordered for the test dataset labels.

The above metrics get the averaged value by evaluating the classification models' performance on the test samples. Hamming loss considers classification quality. Meanwhile, the other metrics evaluate performance in ranking quality.

4.3.2 Result and Analysis

Our experiment is implemented in the following way. In the pre-treatments, LDA removed the words that frequency is one. The ALN-cedm used the same method [26] with Semantic LDA by constructing patent ALN to obtain relations. While, not only had the Semantic LDA excluded the high frequency keywords with weak semantic distinction, but also contracted the Backbone ALN as the relations.

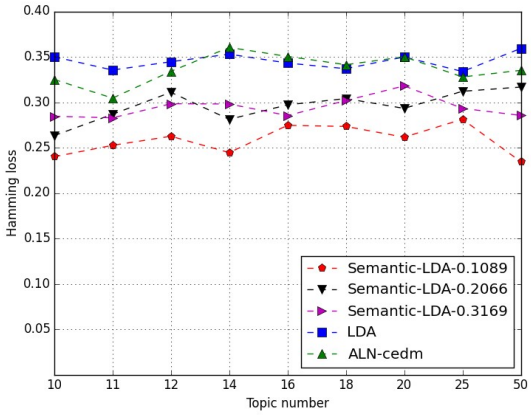
Ten-fold cross-validation is performed on *graphene* dataset. Every experiment repeats 10 times by randomly re-splitting the dataset into the training and the testing sets. We set the topic number from 10 to 50. We defined training dataset's labels set as $\mathcal{L}_{train}$, Φ_{train} is the doc-topic matrix, T is mapping matrix. The problem can be written as a non-negative matrix factorization problem [31], so we estimate T by training dataset, and use it to predict test dataset labels.

The parameters need to be specified for Semantic LDA model are $\alpha = 0.5$, $\beta = 0.1$ and $\gamma = 0.1$. For LDA, $\alpha = 0.5$, $\beta = 0.1$. Through a large number of parameter selection, as Table 7 shows, we use three sets of parameters to build three kinds of average Backbone ALN graph density, and then add these Backbone ALN into Semantic LDA. This experiment compares LDA, ALN-cedm, Semantic-LDA-0.1089 (average Backbone ALN density is 0.1089), Semantic-LDA-0.2066 (average Backbone ALN density is 0.2066), and Semantic-LDA-0.3199 (average Backbone ALN density is 0.3199) in four metrics.

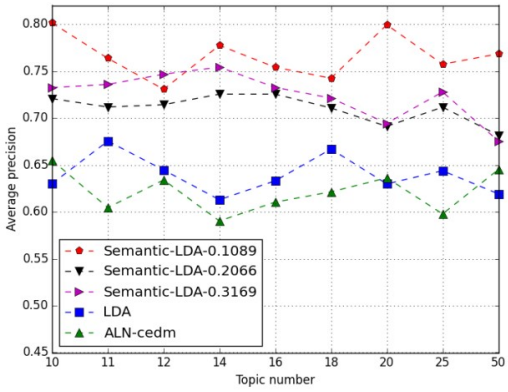
Table 7: The three parameters to build Backbone ALN

μ	λ_1	λ_2	*average Backbone ALN density*
0.29	*0.4*	*0.78*	*0.1089*
0.15	*0.2*	*0.8*	*0.2066*
0.1	*0.27*	*0.85*	*0.3199*

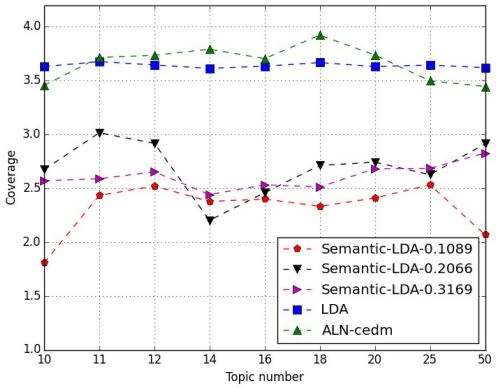
Fig.10 shows the performance of the models in four matrices. For each evaluation metric, "↓" indicates "the smaller the better" while "↑" indicates "the larger the better". Although LDA and ALN-cedm can achieve better some performance in some case, the efficiency of Semantic LDA is generally better than LDA and ALN-cedm in different topic numbers on *graphene* database across all evaluation metrics. Besides, the density of average Backbone ALN has impacts on the efficiency of Semantic LDA. The best is Semantic-LDA-0.1089, and we believe that this value is not fixed and depends on specific dataset. These results indicate that the Semantic LDA topic model proposed in this research is a plausible model that can improve the performance of topic discovery.



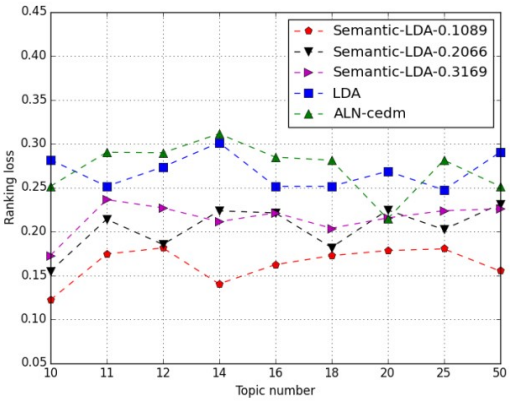
(a) Hamming loss of models shows the mispredicted topic label ratios, the lower ratios of our model prove better performance (i.e.,↓)



(b) Average precision of models are ratios of predicted labels ranked above a particular label. The higher values than baseline demonstrate that our model has satisfying accuracy (i.e.,↑)



(c) Coverage represent predicted labels need to take how many steps to cover instance labels, our model has an obvious advantage in fewer steps (i.e.,↓)



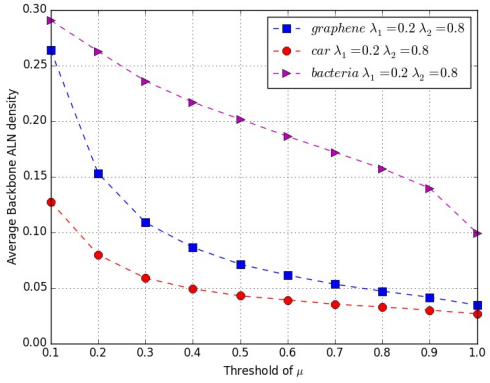
(d) Ranking loss of models evaluate that irrelevant label ranked higher than its relevant one, our model has less loss in most cases (i.e.,↓)

Fig.10 Predictive performance of each comparing algorithm on graphene dataset

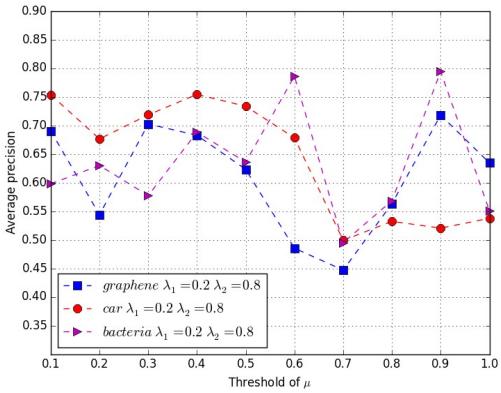
4.3.3 Sensitivity Analysis of Parameters

So far, we have investigated the performance of our model for topic discovery. It is obvious that the Backbone ALN can help to enhance the performance for the topic result. From previous introduction of patent relationships, parameters needed to be specified for building Backbone ALN with various graph densities. In this section, we intend to analyse the effect of the parameters which are used in building Backbone ALN, such as μ, λ_1 and λ_2, and we will show the results of sensitivity analysis on three datasets. Top μ words are selected to expand the relationship in the Domain ALN. λ_1 and λ_2 determine the expand scope of Domain ALN and Patent ALN.

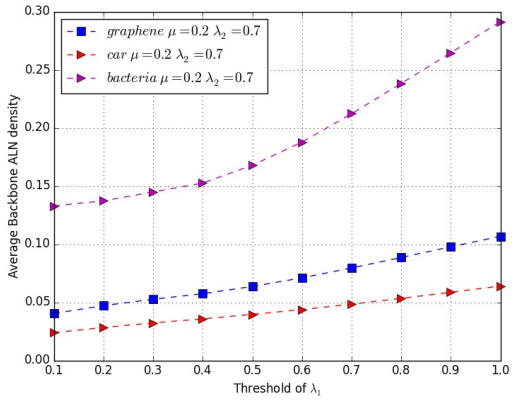
From Fig.11 (a), (b) shows the effect of the patent words, we can observe the changing routes of Backbone ALN density while μ ranging from 0.1 to 1. We set λ_1 as 0.2, λ_2 as 0.9. We use different μ to build Backbone ALN considering top 20% domain relations and top 90% patent internal relationships. According to Fig.11 (a), we can observe that the density of Backbone ALN decline with the increase of μ. As the μ increases, word items grow faster than edge relationships, leads to the average Backbone ALN's density decreases. Fig.11 (b) shows the accuracies as the average Backbone ALN density changing. The average precision is noticeably volatile on three datasets when μ is 0.7. When the value of the average Backbone ALN density is becoming smaller, the average precision is more volatile. We can consider that the performance is disadvantaged by much word and too much word is a noise.



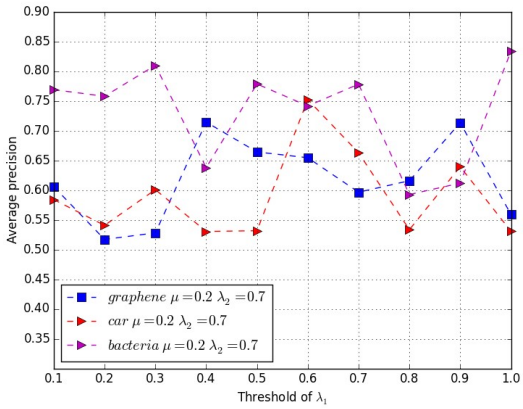
(a) Average Backbone ALN density influenced by percentage of words candidates (μ), horizontal axis represents different range of μ, and vertical axis is density value, the density of Backbone ALN declines as the increase of μ



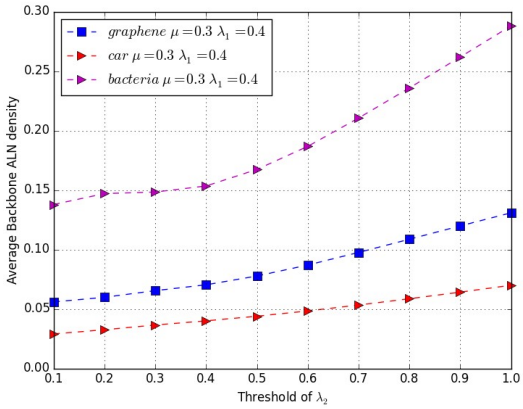
(b) Average precision influenced by μ, the average precision value carves changed with μ fluctuate wildly, so it has an evident effect on precision of topic discovery



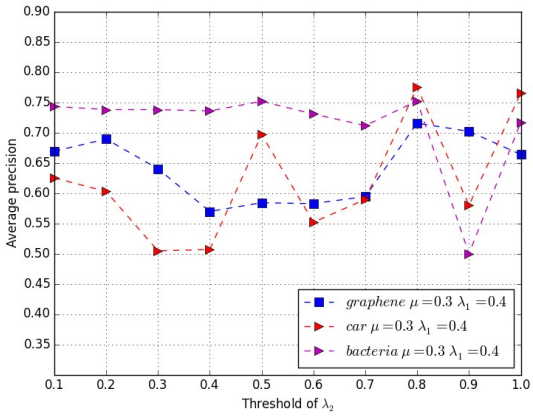
(c) Percentage of domain relations (λ_1) has effect on average Backbone ALN density, horizontal axis represents different range of λ_1, and vertical axis is density value, λ_1 can lead average Backbone ALN density increase as the value increases



(d) λ_1 has influence on average precision, the curves which adjusted by λ_1 are most volatile, it can better prove that the effect of Domain ALN on extending patent semantics is obvious



(e) Percentage of Patent ALN relations (λ_2) has influence on average Backbone ALN density, horizontal axis represents different range of λ_2, and vertical axis is density value, the average Backbone ALN density goes steadily up as λ_2 increases



(f) Average precision influenced by λ_2, the influence on average precision is relative steady on bacteria dataset. From the figure, it is obvious that the changes of precisions are various in different domains

Fig.11 The influence of three parameters such as μ, λ_1 and λ_2 on three datasets in Semantic LDA model

However, in section 4.3.2 Fig.10 (b), in graphene dataset, when the topic number is 12, the average precision of Semantic-LDA-0.1089 is relative high than Fig.11 (b). Even through the density is around 0.1, the average precision results have a great discrepancy between two figures. Because even the average Backbone ALN density is the same, it is hard to ensure that the nodes and relation edges are same, so the average precision is still a little sensitive.

For demonstrating the effect of the Domain ALN scope, we changed the proportion of the λ_1 from top 10% to 100% with the ascending steps of 10%. From the Fig.11 (c), (d), the changes of λ_1 can lead average Backbone ALN density to increase. As the range of Domain ALN increases, more edges are expanded to rebuild Backbone ALN. From Fig.11 (d), the Semantic LDA has slightly worse performance when the value of average Backbone ALN density is 1.

As shown in Fig.11 (e), (f), it is the influence of λ_2 on three datasets, when the number of λ_2 increases, the average Backbone ALN density goes steadily up. We tested model on ten different ranges of patent network. The average Backbone ALN graph density ranges from 0.0290 to 0.2880.

Overall, from Fig.11 (b), (d), (e), in the *graphene* dataset, when the average Backbone ALN density is around 0.1, we can easily get better performance. In three datasets, the average Backbone ALN density is from 0.02 to 0.3. Even we set μ=1，λ_1=1 and λ_2=1, the density cannot reach to 1. Since not all words have edge relations, the highest average Backbone ALN density depends on the strategy of selecting edge relations. We can observe that μ, λ_1 and λ_2 can sensitively influence on the average Backbone ALN density. According to the results, it does not mean that the more relationship, the better average precision. The change of parameters cannot lead to linear or nonlinear relations between densities and precisions. Besides, the average precision is noticeably volatile on different thresholds in the different specific domain. Therefore, it needs some experience to select appropriate parameters to get better performance.

4.4. Semantic LDA Topic Analyzer

All results proved that the Semantic LDA model has better generalization performance. Moreover, the advantage of Semantic LDA is more notable for topic analysis. To investigate the quality of topics discovered by Semantic LDA, we fetch four topics (total topic number is 16) in *graphene* domain for visualization.

As Table 8 shows, each topic shows top 10 words and their conditional probabilities, while we show the top 10 relationships with probabilities. For each topic, we combine top 50 relationships with words to generate semantic topic net.

Topic-1 shows the *battery* semantic topic in the *graphene* domain. From the semantic net, it can easily find the clue that the *battery* how to correlate with *graphene*. The *graphene* as a material of the battery, the key technologies of *battery* topic include {*fullerene*, *graphene*}, {*acid*, *sulfur*}, {*oxide*, *titanium*} etc. The details of the semantic topic are shown in Fig.12.

As we can see from Topic-2, more relationships which are implicated in *screen* topic can be found, as Fig.13 shows, the key point to link *screen* with *graphene* is *nanowire*. The key technologies of *screen* topic include {*composite*, *graphene*}, {*graphene*, *nanowire*}, {*screen*, *print*} etc.

Topic-3 shows the *fiber* semantic topic, except *graphene* node, the degree of *oxide* is five. From the degree value, we can observe that *oxide* is key point in *fiber*. The key technologies of *fiber* topic can be found in {*portion*, *surface*}, {*carbide*, *nitride*}, {*diode*, *emit*} etc. The details of the semantic topic are shown in Fig.14.

Table 8: We selected four topics' top 10 words and relationships with their probability of selected four topics, the first row presents the topic number and quoted topic summary. From our patent sematic topic, we can not only obtain the internal knowledge of the patent content but also gain the global knowledge enriched by domain background knowledge.

Topic-1 "battery" sematic topic

word	*probability*	*relationships*		*probability*
battery	*0.1380*	*battery*	*cell*	*0.0026*
solution	*0.0542*	*fullerene*	*graphene*	*0.0023*
nanocomposit	*0.0351*	*acid*	*sulfur*	*0.0012*
dopant	*0.0216*	*mixture*	*oxide*	*0.0011*
interact	*0.0124*	*active*	*electrode*	*0.0011*
encapsulate	*0.0068*	*battery*	*electrolytic*	*0.0081*
ester	*0.0059*	*battery*	*recharge*	*0.0007*
reactor	*0.0053*	*solid*	*electrolyte*	*0.0068*
hydrophilic	*0.0052*	*pressure*	*temperature*	*0.0061*
lithium-air	*0.0048*	*oxide*	*titanium*	*0.0057*

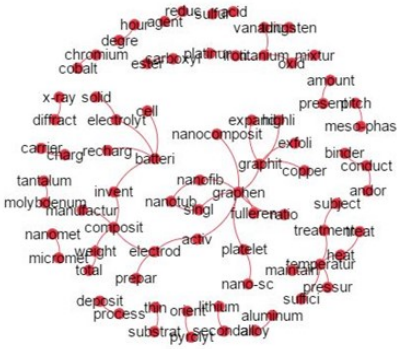


Fig.12 Semantic network of battery topic

Topic-2 "screen" sematic topic

word	*Probabi.*	*relationships*		*probability*
image	*0.0346*	*composite*	*graphene*	*0.0036*
power	*0.0309*	*graphene*	*nanowire*	*0.0035*
hydrogen	*0.0285*	*screen*	*print*	*0.0032*
capacitor	*0.0237*	*dioxide*	*titanium*	*0.0024*
interconnect	*0.0231*	*ultraviolet*	*light*	*0.0021*
screen	*0.2105*	*nanowire*	*screen*	*0.0019*
molecular	*0.0171*	*nitride*	*sulfide*	*0.0009*
aqueous	*0.0123*	*epoxy*	*Polyurethan*	*0.0008*
			.	
mesoporous	*0.0089*	*zinc*	*magnesium*	*0.0018*
curvature	*0.0076*	*ethyl*	*methyl*	*0.0006*

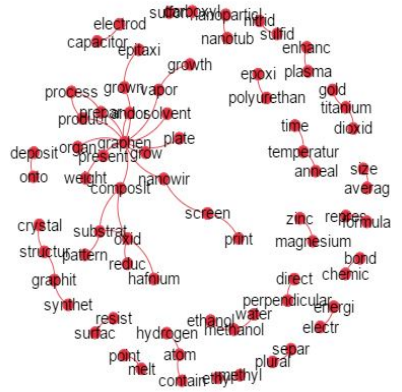


Fig.13 Semantic network of screen topic

Topic-3 "fiber" sematic topic

word	*probability*	*relationships*		*probability*
temperature	*0.0539*	*portion*	*surface*	*0.0028*
fiber	*0.0490*	*carbide*	*nitride*	*0.0019*
compound	*0.0397*	*diode*	*emit*	*0.0017*
liquid	*0.0375*	*drain*	*gate*	*0.0016*
signal	*0.0355*	*germanium*	*silicon*	*0.0025*
anodic	*0.0339*	*polyethylene*	*terephthalate*	*0.0023*
storage	*0.0292*	*degree*	*heat*	*0.0016*
alloy	*0.0217*	*atmosphere*	*inert*	*0.0014*
wire	*0.0212*	*aluminum*	*magnesium*	*0.0013*
aluminum	*0.0190*	*nanotube*	*electrode*	*0.0013*

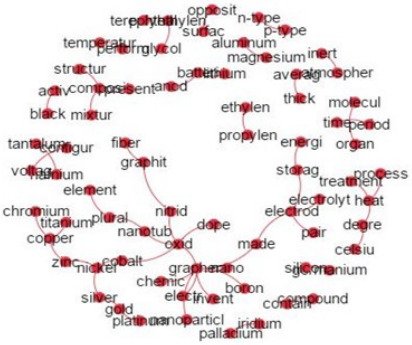


Fig.14 Semantic network of fiber topic

Topic-4 "circuit" sematic topic

word	*probability*	*relationships*		*probability*
radiate	*0.0282*	*graphene*	*oxide*	*0.0098*
adjacency	*0.0234*	*circuit*	*electron*	*0.0053*
flow	*0.0201*	*nitride*	*oxide*	*0.0044*
circuit	*0.0199*	*graphene*	*platelet*	*0.0039*
block	*0.0188*	*insult*	*semiconductor*	*0.0031*
wireless	*0.0145*	*radiate*	*electromagnet*	*0.0027*
polysilicon	*0.0125*	*interconnect*	*electron*	*0.0023*
ultra-low	*0.0116*	*gate*	*voltage*	*0.0019*
colloid	*0.0098*	*dope*	*silicon*	*0.0017*
condense	*0.0086*	*bridge*	*switchable*	*0.0014*

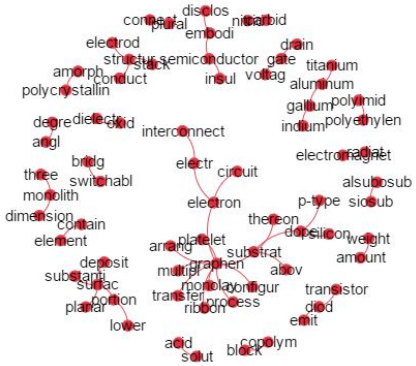


Fig.15 Semantic network of circuit topic

Topic-4 shows *circuit* semantic topic, we can see from this topic, *circuit* is a part of current-carrying, *graphene* is the important role to make up the main material. As Fig.15 shows, the key technologies of *circuit* topic can be found in {*graphene*, *oxide*}, {*circuit*, *electron*}, {*nitride*, *oxide*} etc.

In the case study, the overview *graphene* technology topics can be generated with less effort by Semantic LDA model. Our semantic topics provide an overall understanding about the detail development and current stage of technology in *graphene* domain. The topic results can help to identify critical technologies and their interconnectedness. In a specific domain, these critical technologies can become important breakthrough for enhancing product innovation.

5 Conclusions and Future Work

In this paper, we have proposed a Semantic LDA model to discover semantic topics in a specific domain. In order to overcome the limitations (i.e., low accuracy and poor interpretability) brought by bag-of-words based models, our model has extended LDA by gracefully incorporating ALN considering not only words but also inter-word relationships. First, we have extracted the patent keywords with more semantic distinction ability by considering domain characteristic. Second, we have constructed Domain ALN to mine domain background knowledge while building Patent ALN to represent internal semantics. Then we have built the Backbone ALN with the purpose of balancing the power of both global semantics (Domain ALN) and patent internal semantics (Patent ALN). Finally,

by incorporating patent keywords and relationships from Backbone ALN, we have constructed a Semantic LDA model to discover patent semantic topics. Experimental results have proved that our model can achieve higher accuracy than baseline models. The case study has shown that our semantic topics with higher precision can be easily interpreted. Our work is expected to help identify the key technologies and enhance technology innovation.

In the future, we aim to use the variational inference to improve the inference efficiency of the proposed Semantic LDA model, and extend the model to parallel processing [32][33]. Also, it would be valuable to extend the current model to locate topic changes, to evaluate and analyse the development of the technology topics. Semantic LDA model is fairly generic and can be naturally extend to other complex research area including text mining corpus and image semantics mining [34].

Acknowledgements

The research work reported in this paper was supported in part by the Shanghai Science International Cooperation Project under grant no.16550720400. This work was jointly supported by the National Science Foundation of China under grant no.61471232 and the innovation project of Institute of Computing Technology (ICT), Chinese Academy of Science (CAS).

References

1. Wang W M, Cheung C F. A Semantic-based Intellectual Property Management System (SIPMS) for supporting patent analysis[J]. Engineering Applications of Artificial Intelligence, 2011, 24(8): 1510-1520.
2. Feng L, Peng Z, Liu B, et al. Finding Novel Patents Based on Patent Association[C]//International Conference on Web-Age Information Management. Springer International Publishing, 2014: 5-17.
3. Venugopalan S, Rai V. Topic based classification and pattern identification in patents[J]. Technological Forecasting and Social Change, 2015, 94: 236-250.
4. Chen H, Zhang G, Zhu D, et al. A patent time series processing component for technology intelligence by trend identification functionality[J]. Neural Computing and Applications, 2015, 26(2): 345-353.
5. Noh H, Jo Y, Lee S. Keyword selection and processing strategy for applying text mining to patent analysis[J]. Expert Systems with Applications, 2015, 42(9): 4348-4360.
6. Hu Z, Fang S, Liang T. Empirical study of constructing a knowledge organization system of patent documents using topic modeling[J]. Scientometrics, 2014, 100(3): 787-799.
7. Montecchi T, Russo D, Liu Y. Searching in Cooperative Patent Classification: Comparison between keyword and concept-based search[J]. Advanced Engineering Informatics, 2013, 27(3): 335-345.
8. Park S, Jun S. New technology management using time series regression and clustering[J]. International Journal of Software Engineering and Its Applications, 2012, 6(2): 155-160.
9. Kim K, Khabsa M, Giles C L. Inventor Name Disambiguation for a Patent Database Using a Random Forest and DBSCAN[C]//Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries. ACM, 2016: 269-270.
10. Kang I S, Na S H, Kim J, et al. Cluster-based patent retrieval[J]. Information processing & management, 2007, 43(5): 1173-1182.
11. Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan): 993-1022.
12. Hofmann T. Probabilistic latent semantic indexing[C]//Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1999: 50-57.
13. Supraja A M, Archana S, Suvetha S, et al. Patent search and trend analysis[C]//Advance Computing Conference (IACC), 2015 IEEE International. IEEE, 2015: 501-506.

14. Luo X, Xu Z, Yu J, et al. Building association link network for semantic link on web resources[J]. IEEE transactions on automation science and engineering, 2011, 8(3): 482-494.
15. Tang J, Wang B, Yang Y, et al. PatentMiner: topic-driven patent analysis and mining[C]//Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012: 1366-1374.
16. Venugopalan S, Rai V. Topic based classification and pattern identification in patents[J]. Technological Forecasting and Social Change, 2015, 94: 236-250.
17. Kim G, Park S, Jang D. Technology analysis from patent data using latent dirichlet allocation[M]//Soft Computing in Big Data Processing. Springer International Publishing, 2014: 71-80.
18. Du L, Buntine W, Jin H. A segmented topic model based on the two-parameter Poisson-Dirichlet process[J]. Machine learning, 2010, 81(1): 5-19.
19. Xuan J, Lu J, Zhang G, et al. Topic model for graph mining[J]. IEEE transactions on cybernetics, 2015, 45(12): 2792-2803.
20. Kim Y G, Suh J H, Park S C. Visualization of patent analysis for emerging technology[J]. Expert Systems with Applications, 2008, 34(3): 1804-1812.
21. Che H C, Wang S Y, Lai Y H. Assessment of patent legal value by regression and back-propagation neural network[J]. International Journal of Systematic Innovation, 2010, 1(1).
22. Shih M J, Liu D R. Patent Classification Using Ontology-Based Patent Network Analysis[C]//PACIS. 2010: 95.
23. Chen H, Zhang G, Lu J, et al. A fuzzy approach for measuring development of topics in patents using Latent Dirichlet Allocation[C]//Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on. IEEE, 2015: 1-7.
24. Liu Y, Borhan N, Luo X, et al. Association Link Network Based Core Events Discovery on the Web[C]//Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on. IEEE, 2013: 553-560.
25. Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-620.
26. Luo X, Zhang J, Ye F, et al. Power series representation model of text knowledge based on human concept learning[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2014, 44(1): 86-102.
27. Heinrich G. Parameter estimation for text analysis[J]. University of Leipzig, Tech. Rep, 2008.
28. Forman G. An extensive empirical study of feature selection metrics for text classification[J]. Journal of machine learning research, 2003, 3(Mar): 1289-1305.
29. Griffiths T L, Steyvers M. Finding scientific topics[J]. Proceedings of the National academy of Sciences, 2004, 101(suppl 1): 5228-5235.
30. Zhang M L, Wu L. LIFT: Multi-label learning with label-specific features[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(1): 107-120.
31. Cabral R, De la Torre F, Costeira J P, et al. Matrix completion for weakly-supervised multi-label Image classification[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(1): 121-135.
32. Ng B, Li F W B, Lau R W H, et al. A performance study on multi-server DVE systems[J]. Information Sciences, 2003, 154(1): 85-93.
33. Li F W B, Li L W F, Lau R W H. Supporting continuous consistency in multiplayer online games[C]//Proceedings of the 12th annual ACM international conference on Multimedia. ACM, 2004: 388-391.
34. Yan T, Lau R W H, Xu Y, et al. Depth mapping for stereoscopic videos[J]. International Journal of Computer Vision, 2013, 102(1-3): 293-307.