HIERARCHICAL TOPIC MAP GENERATION FOR EXPLORATORY BROWSING

BY

SHENGLIANG DAI

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Adviser:

Professor Chengxiang Zhai

ABSTRACT

This thesis proposes a novel model for automatically generate topic map for a document corpus with no supervision. We extend a previous approach to discovery of lexical relations from text data to construct a hierarchy of topics. Given a collection of documents, we will generate a set of topics on the fly which will help the user to efficiently navigate through the corpus space and finally land upon the desired document. We use Latent Dirichlet Allocation to generate the top level topics and then leverage paradigmatic and syntagmatic relations between words to construct the hierarchy. We characterize each topic in the hierarchy by a single phrase. Our topic map captures the requirements of user while he/she navigates through the corpus space. Instead of a rigid tree structure, we define links on topic map such that they take user to next desired finer level/related topic based on the history of already visited nodes in map/regions in the corpus. Experiments on DBLP titles datasets show that our topic map can be used very effectively and intuitively by the user to reach to the desired document.

Subject Keywords: Hierarchical topic map, exploratory browsing, lexical relation.

To my parents, for their love and support.

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, Professor Chengxiang Zhai, for guiding me and supporting me during my thesis research. I want to thank him for teaching me hand-to-hand detailed mathematical techniques and inspiring me with his wisdom to make this thesis possible. I sincerely appreciate the help from my colleagues in Text Information Management and Analysis Group (TIMAN) of UIUC, especially Shan Jiang, for helping me adapt her methods in this thesis. and Subham De, for helping me design restricted graphs and set up the experiment and user study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

With the explosive growth of online information, such as news articles, email messages, scientific, literature, government documents, and information about all kinds of products on the Web, search engines have now become essential tools in all aspects of our life. It is estimated that 6 billion user queries were submitted to search engines in October, 2006 alone. Clearly, their effectiveness directly affects our productivity and quality of life. The current search engines are very useful, but they tend to work well only when the user knows sufficiently well about the target web page(s) to formulate an effective query. Unfortunately, often a user does not have any particular target pages in mind or does not know well about the topic to be searched as in exploratory search and informational search. In such cases, it is very difficult for a user to formulate effective queries, and the current search engines generally perform poorly.

When a user is unable to formulate effective queries, browsing would be intuitively very useful because it enables a user to navigate into relevant information (and explore the information space in general) without formulating a query. Indeed, being able to browse the Web through hyperlinks is very important to the web users, and quite often, a user would have to find relevant information by following hyperlinks in the result pages. Had all the hyperlinks been broken, the utility of a search engine would be significantly reduced. In general, when querying does not work well, browsing can be very useful.

Unfortunately, despite the importance of browsing, the current search engines primarily focus on supporting querying and only provide very limited support for browsing, mostly relying on manually created links between pages or directory structures. To support browsing in a more general for arbitrary text collections, we must be able to automatically construct a multi-resolution hierarchical topic map that can cover all the content in a collection.

Developing formal models for information retrieval (IR) is an important problem. the Probability Ranking Principle (PRP) [1] was proposed decades ago and achieved success. It ranks the documents in the collection in order of the probability of relevance to the query. However, PRP is based on the following two problematic assumption: the relevance of a document to a request is independent of the other documents in the collection; the ranking results are browsed in sequence. This is not true in the real world. The sequential browsing assumption limits the interface of search engine and ignores the actions the user could take.

In order to overcome this limitation, PRP-IIR [2] was first proposed which helped to address the independence of documents assumption made in PRP. Dynamic information retrieval system is proposed by [3], which integrates the Query Change Retrieval Model (QCM) [4] and the Win-win search algorithm. It supports dynamic search to different datasets which can be chosen by a user. Y. Zhang and C. Zhai [5] [6] proposed Interface Card Model (ICM) to further generalize PRP-IIR [2].They addressed the limitation of sequential browsing. This model frames the retrieval problem as optimizing a sequence of "interface cards" to be presented to a user in an interactive manner so as to minimize the user's effort while maximizing the gain. The paper [5] focused mainly on the construction of optimal interface card for various kinds of screens. Taking into consideration 2 kinds of elements to be shown in an interface card i.e items(actual documents in the corpus) and tags(topics in a more general sense), they showed that depending on the screen size, their system can automatically generate the optimal number of tags to be shown in the screen e.g more tags than items for a smaller screen. However, the paper does not include generation of tags for the ICM framework. Their experiment used keywords generated by New York Times Most Popular API as tags. But such auto-generated tags may not meet the purpose of the ICM model. Main motivation behind the ICM model is to allow the user to efficiently explore the document corpus space with the help of navigational elements. Navigational elements need to be chosen in a way that they lead to division of corpus space into more focused related areas. Motivation behind generating such hierarchical tags can be more clearly understood by drawing an analogy to a person trying to reach a particular house in a known street. In this case person simply goes to particular street and then looks around for the specific house. Similarly ICM model integrated with hierarchical tags will help an user to find few specific items from a huge corpus. The user can then manually examine these few documents based on specific needs. Thus the purpose of this work is to integrate a hierarchical topic generation model with Interface Card Model to optimize document retrieval system.

In this thesis, we propose a new approach to solve this problem. The problem of constructing a topic map is not new, and many approaches can be potentially used to achieve the goal, including, e.g., clustering of documents [7, 8, 9, 10, 11, 12] or using topic models [13, 14, 15, 16, 17, 18]. A key novelty of our approach as compared with the previous approaches is that we generate a hierarchical topic map by first mining two fundamental lexical relations between words (i.e., paradigmatic and syntagmatic relations) from the collection and then using these relations to construct a hierarchical map that satisfies multiple desirable properties (See Chapter 3) that a topic map should satisfy in order to effectively support browsing. We conduct the experiment on DBLP (computer science bibliography) paper titles because it is from the domain of computer science and easy to be evaluated.

As demonstrated in our experiments, the construction of our topic maps is on the fly and the subtopics for the topics can be clearly identified and easily interpreted. By conducting the user study with unrelated topic identification test, we can know if the topic map makes sense to users. We have 80.8 % accuracy on average (the percentage of correctly answered test questions).

The main contributions of this thesis are as follows:

1. We identify multiple requirements of user for exploratory browsing.

2. We propose a novel way of constructing topic map from a corpus using paradigmatic and syntagmatic relations between words that can satisfy the identified requirements.

3. We propose a novel method for extracting syntagmatic and paradigmatic relation to phrases.

4. We propose to evaluate a topic map using a novel intrusion task called unrelated topic identification and show promising results of the proposed topic map construction method.

The rest of the thesis is organized as follows. We first briefly review related work in Chapter 2. We discuss user requirements for exploratory browsing in Chapter 3. We discuss our methodology in Chapter 4. We present experimental results in Chapter 5. In Chapter 6, we will give a conclusion for the experiments and future work.

CHAPTER 2

RELATED WORK

In this chapter, we make an overview of relevant methods and concepts for topic models. In section 2.1, we discuss the topical keyphrase extraction and ranking. In section 2.2, we introduce the state-of-the-art topic modeling algorithms.

2.1 Topical Keyphrases Extraction and Ranking

The keyphrases can be extracted using Natural Language Processing techniques or statistical language model[19]. The state-of-the-art unsupervised topical keyphrases approaches extract uni-grams from the documents and rank them for each topic, and finally combine them to keyphrases [20, 21]. X. Ren et al. [22] use a graph-based framework to derive novel measures on *phrase semantic commonality* and *pairwise distinction*. They perform distantly-supervised phrase segmentation on documents and select salient phrases to represent each document.

2.2 Topic Modeling

Topic modeling approaches such as Latent Dirichlet Allocation (LDA) [23] take the documents as input, and model them as a mixture of multiple topics, and output the distribution of topics. However such topics are not hierarchical in nature. Hierarchical Pachinko Allocation model hPAM [24] and hLDA [25] can mine hierarchical topics from a given corpus. But these methods adhere to a very strict tree structure i.e each topic has a set of child topics and the number of documents covered by child topic is less than that covered by parent topic. These topic maps contain only vertical links i.e user can move from parent topic to child topic. While this is one vital movement pattern for the user that must be supported by a topic map, there may be other movement requirements for the user as well. For example, say Sandy was reading an IR paper. Now she realizes that this paper and all other related papers are using an algorithm from the ML domain, which she is not aware of. So she now desires to visit the ML domain and explore papers related to that algorithm. None of the

existing topic models can support such sort of exploratory browsing.

Wang et.al. [26] propose a new kind of topic model which contains horizontal links in addition to vertical links. They however use query logs instead of documents to construct the topic map. Also, in their map they have defined horizontal links between nodes that have high document overlap. From exploratory point of view, such links do not seem to be meaningful. When user wishes to navigate away from current topic node, that means she is not interested in documents under current node. So there is no point of taking user to a node that is still covering a majority of the same nodes. Rather we should aim to take the user to a node that is highly related to the previous one, but having minimal document overlap. If we imagine the corpus as a vast continuous space, we should take user to neighboring regions in that space. Cathy [27] uses a generative model to generate the topic hierarchy. Here, each topic is modeled as a word co-occurrence graph. Unlike LDA which considers each topic as word occurrence, here they generate word co-occurrence graph for subtopics given the parent topic. The main difference of this work from our work is that it constructs topic map as a static tree like structure.They are interested in capturing concept hierarchy in the given corpus, but our goal is to dynamically support browsing of users through the corpus.

2.3 Existing topic map models

Ahmed et. al. [28] introduces the ISO international standard Topic Maps. The topic maps paradigm describes a way in which complex relationships between abstract concepts and real-world resources can be described and interchanged using a standard XML syntax. Kannan [29] explores how topic map incorporates the traditional techniques and what are its advantages and disadvantages in several dimensions such as content management, indexing, knowledge representation, constraint specification and query languages in the context of information retrieval.

The differences between these two articles with this thesis are the construction and representation of the topic map. [29] consider university faculty profile as a subject of study. The picture of the structure of their topic map model is shown in Fig. 2.1. The three fundamentals of their model of topic map: topics, associations and occurrences. They represent topic map as a network of nodes, not a tree hierarchy.

From Fig. 2.1, the topic map is divided into topic space and resource space through topic-to-topic and topic-to-resource relationships. This partitioning helps merging and extending topic map alone without affecting information resources. Also, information resources can be

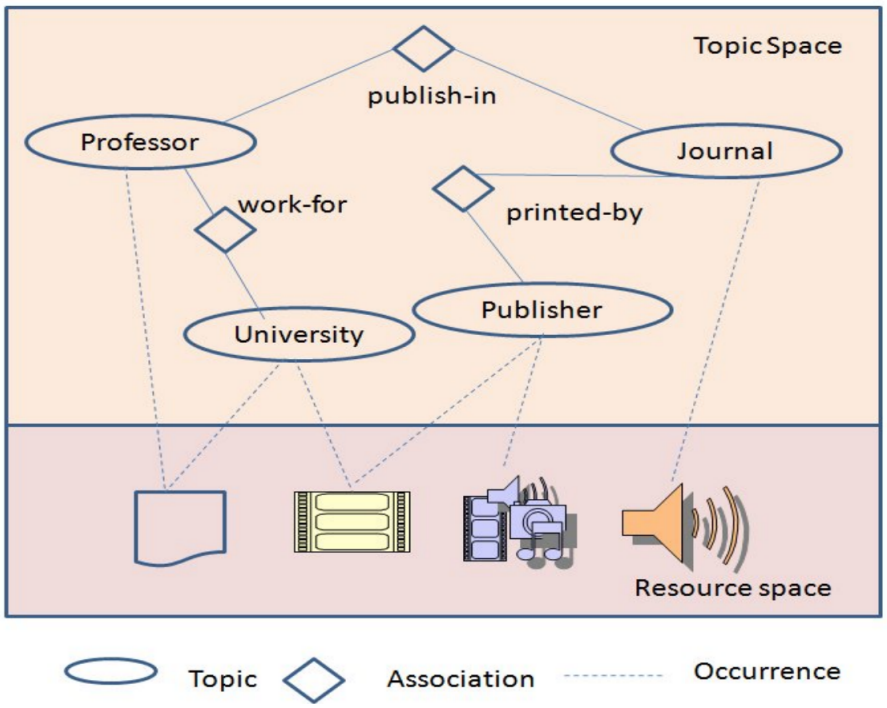

Figure 2.1: Existing topic map model

updated without disturbing topic map. This is an advantage of their model because this model can easily show the relationship and association between topics. Compared to our model, we do not consider the association and each node is represented by phrases.

In this thesis, we represent the topic map level by level, which means that the user would choose a node and dive deeper to the next level from previous node recursively. [28], [29] use a standard XTM syntax to represent the topic map in Fig. 2.2. The XTM syntax is a syntax based on XML, XLink, and URIs. In this thesis, we currently do not incorporate this syntax with our model, which is a disadvantage when we want to represent the topic maps with text or serialize the topic map.

```
  <topic id="#professor">
  <baseName>
  <baseNameString>Professor</ba
seNameString>
  </baseName>
  </topic>
  <topic id="#rajkumar-kannan">
  <instanceOf>
  <topicRef
xlink:href="#professor"/>
  </instanceOf>
  <baseName>
  <baseNameString>Rajkumar
Kannan</baseNameString>
  </baseName>
  </topic>
```

Figure 2.2: XTM syntax of topic map

CHAPTER 3

PROBLEM DEFINITION

The goal of this thesis is to develop a topic map construction method to build a topic map from a large collection of documents without supervision. The input is a collection of documents in a specific domain. The output is a topic maps with hierarchy. In this section, we briefly introduce basic concepts and components as preliminaries. First, we give definitions of the user requirements for the topic map. Second, we define the topic map.

3.1 User Requirements

As discussed in the earlier section, none of the existing methods are efficient enough to allow the user to comfortably explore the corpus space. In order to support such smooth browsing, we first need to clearly identify the user necessities. Or more explicitly we need to answer these 2 questions:

Given that the user is in a particular node in the topic map, why will he/she want to move to another node rather than directly viewing documents under the current node? Which nodes should we show to the user next given the history of already visited nodes.

Motivated by this question, we have identified 4 distinct scenarios that cover various requirements while exploratory browsing.

- Number of documents under current node are too high or rather the topic of current node is too general. In such a case, user will want to explore more finer topic nodes of the map. Perceived cost of directly analyzing the documents is higher than visiting another node in the map. This requirement directly validates the existence of vertical links in topic map.

- User has landed on a region in the corpus which is very close to the desired region. After analyzing documents, user realizes she is in a region which covers desired topic very closely or as a minor theme. e.g object-oriented database systems to relational database systems. In that case, we need to take user to neighboring region in corpus.

- User needs to go to a region which is methodologically related to the current node. e.g Information Retrieval(IR) uses methods from Machine Learning(ML) domain.
- While taking user to neighboring regions, we should keep in mind the broader picture that the user is interested in. For example say user U has taken this path on the topic map :

  relational database $\rightarrow$ relational database systems.

  Now if U wants to see related topics to relational database systems, we should not show topics like object-oriented database systems because the user has already made it clear that her broader interest is relational database.

Based on these requirements, we define a topic map as follows.

3.2 Topic Map

We define topic map as **V**. Each node in the map **V** is defined by

$$V = (L, \mathbb{R}, \mathbb{F}) \tag{3.1}$$

where L is the label of node. We use a single phrase as node label. $\mathbb{R}$ is the set of related topics, $\mathbb{F}$ is the set of finer/child topics e.g.

(database, {data,dbms,..}, {object-oriented database,..})

$$\mathbb{R} = \{V_{R_1}, ..., V_{R_i}, ..\}$$

$$\mathbb{F} = \{V_{F_1}, ..., V_{F_i}, ..\}$$

V_{R_i} and V_{F_i} are other nodes in the topic map.

CHAPTER 4

METHODOLOGY

In this chapter, we describe the proposed method. We first present the full procedure of our proposed **Topic Map**. Then we introduce each of them in following sections.

1. Introduce the concept of paradigmatic relations and syntagmatic relations
2. Use topic model (LDA) to extract the root topics.
3. Extract paradigmatic and syntagmatic relations to extend the topic hierarchy.
4. Apply random walks on restricted graphs

4.1 Paradigmatic Relations

Two words are said to be paradigmatically related if they share a lot of context. For example, since car and vehicle are used in the same meaning, they are highly paradigmatic word. By applying relation on academic corpus, we see words like *database* and *data mining* are highly paradigmatically related.So it is apparent we can use these relations to meet 2^{nd} user requirement identified in the earlier section i.e data mining is a related topic to database.

4.2 Syntagmatic Relations

Two words are said to be syntagmatically related if they co-occur a lot in the same context. e.g "relational" and "database" are highly syntagmatic words because of lot of occurrence of phrase *relational database*. "relational" shares syntagmatic backward relation with "database" as it occurs before "database". On the other hand, "database" shares syntagmatic forward relation with "relational". We will use these relations to meet 1^{st} user requirement identified in the earlier section i.e "relational database" is a finer topic to database.

4.3 Extracting Root Topics

The first step is to generate the root topics so that we can extend the hierarchy based on these root topics. To generate the root topics, we choose Latent Dirichlet Allocation (LDA) [23] as the method.

We first introduce the concept of LDA, then we discuss its advantages as well as disadvantages.

The input is a collection of documents $D = \{d_1, d_2, \cdots, d_n\}$. We first apply LDA on the collection to generate a set of topics as the root of topic map, $T_i = \{t_{i1}, t_{i2}, \cdots, t_{im}\}$.

Latent Dirichlet Allocation [23] is an unsupervised machine learning method for collections of discrete data such as text corpora.

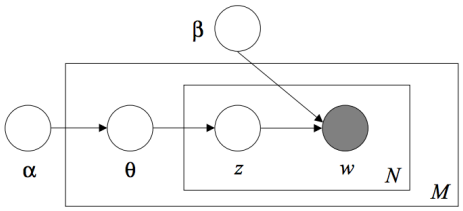

Figure 4.1: Graph model of LDA.

In figure 4.1, the boxes in the graph are plates meaning replicates. The outer plate represents documents of M. The inner plates represents the repeated N choices of topics and words in one document. The nodes α and β are hyper parameters of the text collections.

The generation of LDA works as follows. For each document d, a multinomial distribution θ is generated from the a prior distribution Dirichlet α. Then for each word w_i in document d, a topic z_i is generated from multinomial distribution parameterized by θ. The word w_i is generated from topic z_i specific multinomial distribution parameterized by β. [23] show that LDA can not find semantic low dimensional representations, but can classify and filter documents effectively.

Overall, the advantages of LDA model is its modularity and extensibility. LDA can be readily embedded in a more complex model. There are numerous possible extensions of LDA. For example, LDA can be extended to continuous data or non-multinomial data.

4.4 Extending Hierarchy

We now borrow methods from Jiang et. al.[30], and thus the following presentation is entirely based on Jiang et. al. [30], to extract paradigmatic and syntagmatic relations. Then we

define restricted graph in section 4.7 to generate phrases labels. We will represent text data as a word adjacency graph. In the next subsection, we will first introduce word adjacency graph.

4.4.1 Notations

A vocabulary set V is denoted as $V = \{v_1, v_2,$
$\ldots, v_N\}$, where $v_i (1 \leq i \leq N)$ is a unique word in the dataset. We can treat sentence, paragraph or even document as an individual sequence, then construct sequence-based adjacency graph.

A sequence s_i is represented as $< v_{i1}, v_{i2}, \ldots, v_{il} >$, where $v_{ij} \in V, 1 \leq j \leq l$, l is the length of sequence s_i. A sequence set S with n sequences is denoted as $S = \{s_1, s_2, \ldots, s_n\}$.

Given two sequences s_p and s_q ($s_p =< v_{p,1}, v_{p,2}, \ldots, v_{p,l_p} >, s_q =< v_{q,1}, v_{q,2}, \ldots, v_{q,l_q} >$), then $s_p \subseteq s_q$ holds (s_p is a subset of s_q) if $l_p \leq l_q$, and there exists an integer $r (1 \leq r \leq l_q - l_p + 1)$ such that $v_{p,1} = v_{q,r}, v_{p,2} = v_{q,r+1}, \ldots, v_{p,l_p} = v_{q,r+l_p-1}$.

4.4.2 Constructing Adjacency Graph:

Nodes in the adjacency graph G_k are words in the text data. Therefore, we use the V_k to denote node set. We can derive a series of adjacency graphs from a sequence set by taking different kinds of adjacency mainly *immediate* and *skipped* adjacency. e.g In the sequence w_1, w_2, w_3 , w_1 and w_2 have immediate adjacency while w_1 and w_3 have 1-skipped adjacency.

4.5 Random Walks on Adjacency Graph

Given an adjacency graph G, we define forward walking and backward walking. Suppose we have a set of edges $(v_{r_1}, v_{r_2}), (v_{r_2}, v_{r_3}), \ldots, (v_{r_l}, v_{r_l+1})$ in G. A forward walking for $v_{r_1} \rightarrow v_{r_2} \ldots \rightarrow v_{r_l} \rightarrow v_{r_l+1}$ is to visit $v_{r_1}, v_{r_2}, \ldots, v_{r_l+1}$ sequentially. A backward walking for $v_{r_1} \dashrightarrow v_{r_2} \ldots \dashrightarrow v_{r_l} \dashrightarrow v_{r_l+1}$ is to visit $v_{r_l+1}, \ldots, v_{r_2}, v_{r_1}$ sequentially. Then we can build a transition matrix with the nodes of the graph and mine relations between different words and phrases.

We define the $l - step$ forward walking $v_i \rightarrow^l v_j$ as $\{v_i \rightarrow v_{r_1} \rightarrow v_{r_2} \ldots \rightarrow v_{r_l-1} \rightarrow v_{r_j} | (v_i, v_{r_1}), (v_{r_1}, v_{r_2}), \ldots, (v_{r_l-1}, v_{r_j}) \in E\}$, and an $l - step$ backward walking $v_i \dashrightarrow^l v_j$ as $\{v_i \dashrightarrow v_{r_1} \dashrightarrow v_{r_2} \ldots \dashrightarrow v_{r_l-1} \dashrightarrow v_{r_j} | (v_{r_1}, v_i), (v_{r_2}, v_{r_1}), \ldots, (v_{r_j}, v_{r_l-1}) \in E\}$

The probability of an $l - step$ forward walking from v_i to v_j in a graph G is defined as:

$$P_G(v_i \rightarrow^l v_j)$$

Similarly, the probability of an $l - step$ backward walking from v_i to v_j in a graph G is defined as:

$$P_G(v_i \dashrightarrow^l v_j)$$

Given the adjacent matrix A of graph G, we define $A(i,j) = w[(v_i, v_j)]$, where $w[(v_i, v_j)]$ is the edge weight between (v_i, v_j) in G. Then we have two diagonal matrix D_F and D_B:

$$D_F(i,i) = \frac{1}{\sum_{j=1}^{|V|} A(i,j)} \tag{4.1}$$

$$D_B(j,j) = \frac{1}{\sum_{j=1}^{|V|} A(j,i)} \tag{4.2}$$

Both $D_F(i,j)$ and $D_B(j,i)$ will be 0 if $i \neq j$. Then we define the forward and backward walking transition matrix T_F and T_B as:

$$T_F = D_F A \tag{4.3}$$

$$T_B = D_B A^T \tag{4.4}$$

It is obvious that the 1-step forward walking transition probability $P_G(v_i \rightarrow^1 v_j) = T_F(i,j)$ and the 1-step backward walking transition probability $P_G(v_i \dashrightarrow^1 v_j) = T_B(i,j)$. Based on this, we can compute

$$P_G(v_i \rightarrow^l v_j) = \sum_{v_r \in V} P_G(v_i \rightarrow^{l-1} v_r) P_G(v_r \rightarrow^1 v_j) = T_F^l(i,j) \tag{4.5}$$

$$P_G(v_i \dashrightarrow^l v_j) = \sum_{v_r \in V} P_G(v_i \dashrightarrow^{l-1} v_r) P_G(v_r \dashrightarrow^1 v_j) = T_B^l(i,j) \tag{4.6}$$

4.6 Mining Relations

In this section, we will introduce how to discover paradigmatic and syntagmatic relations using "round trip" random walks.

4.6.1 Mining Paradigmatic Relations

Paradigmatic relations can be mined by clockwise and anti-clockwise "round trip" random walks. The first step is to find all the common neighbours that are reachable from v_i and v_j. In Figure 4.2, v_u and v_w are the common neighbours that are reachable from v_i and v_j. And the clockwise circle trip is $v_i \rightarrow^l v_w \dashrightarrow^l v_j \dashrightarrow^l v_u \rightarrow^l v_i$. The anti-clockwise circle trip in Figure 4.3 is $v_i \dashrightarrow^l v_u \rightarrow^l v_j \rightarrow^l v_w \dashrightarrow^l v_i$.

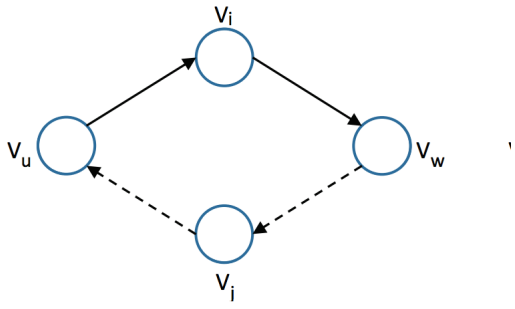

Figure 4.2: Clockwise circle trip for paradigmatic relation mining

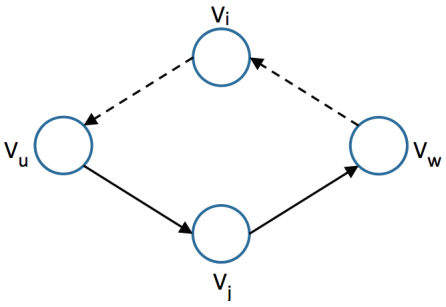

Figure 4.3: Anti-clockwise circle trip for paradigmatic relation mining

We denote the probability of taking l-step circle trip with ends of v_i and v_j in both clockwise and anti-clockwise in graph G as $P_G(v_i \bigcirc^l v_j)$. We denote $\{v_u | P_G(v_u \rightarrow^l v_i) > 0 \wedge P_G(v_u \rightarrow^l v_j) > 0\}$ as U and $\{v_w | P_G(v_i \rightarrow^l v_w) > 0 \wedge P_G(v_j \rightarrow^l v_w) > 0\}$ as W. Then

$$\begin{aligned} P_G(v_i \bigcirc^l v_j) = \sum_{v_u \in U} \sum_{v_w \in W} & P_G(v_i \rightarrow^l v_w) P_G(v_w \dashrightarrow^l v_j)) \cdot \\ P_G(v_j \dashrightarrow^l v_u) P_G(v_u \rightarrow^l v_i) \cdot \sum_{v_u \in U} \sum_{v_w \in W} & P_G(v_i \dashrightarrow^l v_u) \cdot \\ & P_G(v_u \rightarrow^l v_j) P_G(v_j \rightarrow^l v_w) P_G(v_w \dashrightarrow^l v_i) \end{aligned} \tag{4.7}$$

The disadvantage of this is that it is easy to cause bias on frequent words (or stop words) because when $|U|$ and $|W|$ are large, $P_G(v_i \bigcirc^l v_j)$ will also be large. Therefore, we normalize $P_G(v_i \bigcirc^l v_j)$ by the number of all possible circle trips in a unique direction, which is $|U||W|$. $P_G(v_i \bigcirc^l v_j)$ reflects how similar the context of v_i and v_j, if the value of $P_G(v_i \bigcirc^l v_j)$ is large, that means v_i and v_j have high paradigmatic relation. For example, if v_i is "Monday", v_j has very high probability to be "Tuesday" or "Wednesday".

The random walker could choose different l to complete the circle trip, we define α_l to be the prior probability to choose l and:

$$\sum_{l=0}^{s} \alpha_l = 1 \tag{4.8}$$

where s is the maximum step allowed in the circle trip. The normalized score for $P_G(v_i \bigcirc^l v_j)$ is:

$$\sum_{l=0}^{s} \frac{P_G(v_i \bigcirc^l v_j)\alpha_l}{|U||W|}$$

Combining different adjacency graphs induced from the same datasets, we use $Pr(v_i, v_j)$ to measure paradimatic relation.

$$Pr(v_i, v_j) = \sum_{k=1}^{K} \beta_k \sum_{l=0}^{s} \frac{P_G(v_i \bigcirc^l v_j)\alpha_l}{|U||W|} \tag{4.9}$$

where β_k is the prior probability to choose graph k to perform random walks, $\sum_{k=1}^{K} \beta_k = 1$.

4.6.2 Mining Syntagmatic Relations

The syntagmatic relation can be captured by co-occurrence between words. If v_i and v_j co-occur a lot and v_i always occur before v_j, then the probability of a forward walking from v_i to v_j and a backward walking from v_j to v_i is likely to be high. The advantage of round trip is that it will eliminate the domination of "popular" nodes. A node is a "popular" node if it has a large number of out-links and in-links. Therefore, it is easy to reach a "popular" node, but it is difficult to complete a round trip because there are too many out-links to choose.

If the task for a random walker is to take an $l - step$ forward walking from v_i to v_j and then return to v_i by an $l - step$ backward walking, where l should be less than s and can be

chosen in advance with a probability of α_l, then the total probability for the random walker to reach v_j as the destination (considering all the possible paths) is:

$$P_G^s(v_i \rightarrow v_j) = \frac{\sum_{l=1}^{s} P_G(v_i \rightarrow^l v_j) P_G(v_j \dashrightarrow^l v_i)\alpha_l}{\sum_{v'_j \in V} \sum_{l=1}^{s} P_G(v_i \rightarrow^l v'_j) P_G(v'_j \dashrightarrow^l v_i)\alpha_l} \tag{4.10}$$

$$\propto \sum_{l=1}^{s} P_G(v_i \rightarrow^l v_j) P_G(v_j \dashrightarrow^l v_i)\alpha_l$$

Similarly, we can define a backward-first walking, which means the first step is to walk backward. The probability of successfully reaching v_i is:

$$P_G^s(v_i \leftarrow v_j) = \frac{\sum_{l=1}^{s} P_G(v_i \dashrightarrow^l v_j) P_G(v_j \rightarrow^l v_i)\alpha_l}{\sum_{v'_j \in V} \sum_{l=1}^{s} P_G(v_i \dashrightarrow^l v'_j) P_G(v'_j \rightarrow^l v_i)\alpha_l} \tag{4.11}$$

$$\propto \sum_{l=1}^{s} P_G(v_i \dashrightarrow^l v_j) P_G(v_j \rightarrow^l v_i)\alpha_l$$

If $P_G^s(v_i \rightarrow v_j)$ is high, then v_i has the high probability to be the predecessor of v_j. For example, v_i is "more", v_j is "than". if $P_G^s(v_i \leftarrow v_j)$ is high, then v_i has the high probability to be the successor of v_j. For example, v_j is "much", while v_i is "more".

4.7 Random Walks on Restricted Graphs

Previous subsections give us ways to extract paradigmatic and syntagmatic relations to words like *data* , *mining* etc. But in our topic map, each node is represented by a phrase label like *data mining*. So while defining finer and related topics for such a node, we need to extract relations to a phrase. Naive solution may be :

1. Apply paradigmatic relation to data and mining separately and take the union. But it leads to some meaningless phrases like *lifecycle mining*, *data grand* etc

2. Apply syntagmatic forward relation to mining and apply syntagmatic backword relation to data. Again it leads to uninteresting phrases like *uncertain data mining*, *data mining periodic* etc

Instead we propose a new solution which constructs a new graph called restricted graph. Restricted adjacency graph for given phrase p, $G_p$is defined as

- for $w \in p$, consider only the context where entire p occurs.
- for $w \notin p$, consider all context.

Now we apply naive methods discussed earlier on this restricted graph. For the given phrase, *data mining*, it gives us interesting neighbor topics like *pattern mining*, *opinion mining* and child topics like *data mining techniques*, *privacy-preserving data mining* etc.

4.8 Constructing Hierarchy

Hierarchy construction algorithm is given in 4.4. We use inverted index technique to find the documents covered by a given phrase P. We use parent history h to find previous nodes and apply paradigmatic relation accordingly so that we can avoid showing repetitive topic nodes to the user. In the experiments conducted we have chosen the value of *threshold* to be 10 [1].

```
Algorithm:Hierarchy Construction(Phrase p,ParentHistory h)

 If(DocsCovered(p) < threshold)
        return Docs;
 else
        words = phrase.component()
        for(words[i] not in h)          ////re-defining topic map based on history
                neigborTopics = U (paradigmaticRelation(words[i], context=phrase))
        finerTopics = (syntagmaticFwdRelation(words[last_word], context=phrase))
                              U (syntagmaticBckRelation(words[first_word], context=phrase))
```

Figure 4.4: Hierarchy algorithm

[1]The code of this thesis is available at at https://github.com/daishengliang/topic-map

CHAPTER 5

EXPERIMENTS

In this section, we will apply the proposed method to construct topic map from text corpora in computer science articles. We use DBLP paper titles as the dataset. We perform qualitative study and user study in the end. The experiment shows that the construction of our topic maps is on the fly and the subtopics for the topics can be clearly identified and easily interpreted.

5.1 Experimental Settings

In this section, we introduce the datasets and the experiment results. Then we describe our evaluation: we conduct a user study with "intruder detection" tasks to evaluate hierarchy quality.

5.2 Datasets

We run our model on DBLP datasets:

- **DBLP** We collected a set of titles of computer science papers from DBLP [1] in the areas related to Databases, Data Mining, Information Retrieval, Machine Learning , Natural Language Processing etc. We removed the stop-words from the titles. The set has 1.9M titles, 152K unique terms, and 11M tokens. So the root topics for this dataset is computer science. The various subtopics under "computer science" is data mining, machine learning, information retrieval, database and theory.

  The reason we choose this dataset is that the paper titles contain abstract information for the paper, which is easy for setting the experiment.

[1] The datasets are available at at http://illimine.cs. illinois.edu/cathy

database	
Related topics	Finer topics
data	database systems
dbms	database system
recommender	database management
databases	database design
information	database schemes
retrieval	object-oriented database
production	relational database
expert	distributed database
management	multiprocessor database
storage	federated database

Table 5.1: Nodes at second level for "database"

5.3 Experiments and Performance Study

We run our algorithm on the DBLP titles dataset, and generate a static version of the topic map. As mentioned earlier, our algorithm constructs topic maps on the fly. Nodes as well as links seen by user vary based on requirements, history etc. Here we present one such version of the map, as seen by our human evaluator.

5.3.1 Qualitative Results

A static version of topic map constructed by our method is shown in Table 5.1, 5.2, 5.3, 5.4, 5.1. As it can be clearly seen, the subtopics for the topics can be clearly identified. They can be very easily interpreted e.g object-oriented database for database. Similarly, related topics are also very close to the original topic e.g classification to clustering.

The words under the "Related topics" column are words generated by **paradigmatic relations**. The words under "Finer topics" column are words generated by **syntagmatic relations**.

5.3.2 Evaluating Topic Analysis

Topic map evaluation has similar difficulties to information retrieval evaluation. The reason is that there is usually not one true answer in both cases, and the evaluation metrics heavily depend on human issuing judgments.

Log-likelihood and model perplexity are two common evaluation measures used by language models, and they can be applied for topic map evaluation in the same way. Both are

clustering	
Related topics	Finer topics
clusters	spectral clustering 4
learning	subspace clustering
classification	document clustering
outlier	k-means clustering
retrieval	hierarchical clustering
mining	density-based clustering
collections	agglomerative clustering
algorithm	clustering algorithm
categorization	clustering categorical
cluster	clustering ensembles

Table 5.2: Nodes at second level for "clustering"

classification	
Related topics	Finer topics
categorization	text classification
learning	multi-label classification
retrieval	genre classification
movie	sentiment classification
association	document classification
clustering	classification rules
mining	classification unlabeled
answering	semi-supervised classification
classifier	associative classification
classifiers	classification accuracy

Table 5.3: Nodes at second level for "classification"

information retrieval	
Related topics	Finer topics
document retrieval	information retrieval systems
text retrieval	interactive information retrieval
image retrieval	information retrieval workshop
passage retrieval	information retrieval question
ad-hoc retrieval	information retrieval abstract
information management	cross-language information retrieval
information extraction	model information retrieval
information integration	music information retrieval
information systems	distributed information retrieval
information filtering	intelligent information retrieval

Table 5.4: Nodes at third level for "information retrieval"

predictive measures, meaning that held-out data is presented to the model and the model is applied to this new information, calculating its likelihood. If the model generalizes well to this new data (by assigning it a high likelihood or low perplexity), then the model is assumed to be sufficient.

5.3.3 Topic Intrusion User Study

To quantitatively measure topic map quality, we have conducted user study. We asked 4 computer science graduate students to judge the results since they are knowledgeable in this domain.

In order to evaluate the quality of generated topics, we use a modified version of tasks from Chang et al. [31]. The original task is called topic intrusion, which tests the quality of generated map by adding a intruder topic. During the task, for a particular topic T, users are given N candidate sub-topics. $N-1$ of them are true sub-topics, while one is intruder. Users are asked to select the intruder sub-topic. In our framework, it is not possible to directly use this task since child topics are all derived by adding words through syntagmatic relations to the parent topic. e.g object-oriented database to database. So due to the presence of parent topic node phrase in child node phrases, they may be easily identified by evaluator leading to 100% accuracy which may be misleading. Instead, we develop a new task called **unrelated topic identification** to evaluate our model. In this task, for topic T , we give user N options. $N-1$ of them are related. We use finer topics for current topic T and related topic set $\mathbb{R}$ as candidate set for generating these $N-1$ choices. The last choice is chosen from distant topic i.e which is bit far in the topic map. We ask the user to identify most unrelated topic to the given topic. Task is depicted in the figure 5.2.

For user study, we set N to be 4 and asked each participant to answer 30 questions with this format. We compared participants' answers with our solution and calculated the percentage of correct answered questions for each participant. We got 80.8% accuracy on average (the percentage of correctly answered questions). The table in Fig 5.3 shows result for each individual participant.

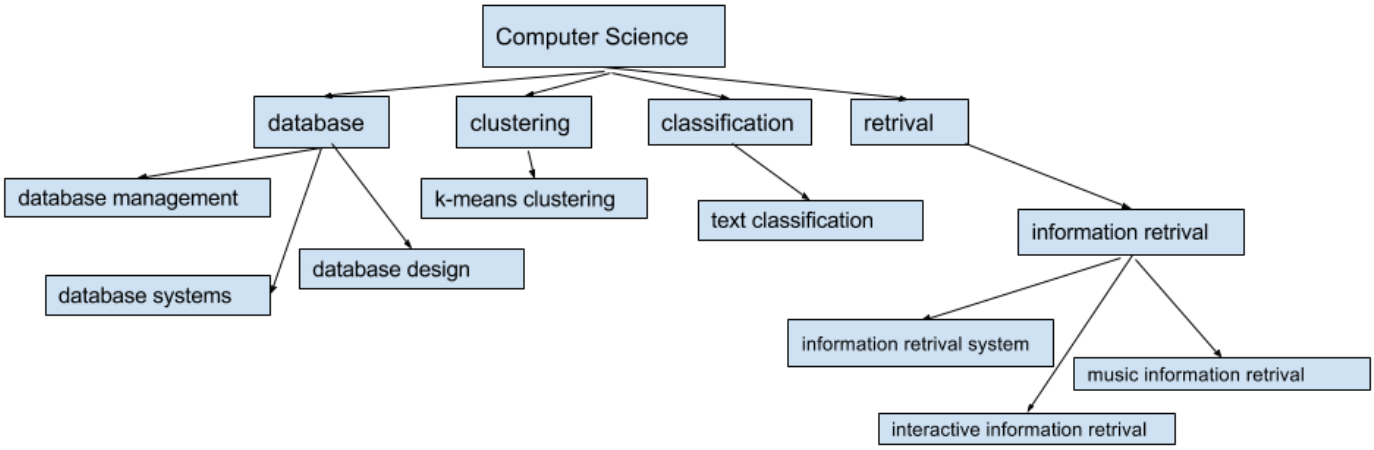

Figure 5.1: Examples of topic map

Topic	Choice 1	Choice 2	Choice 3	Choice 4
clustering	categorization methods	database applications	categorization algorithm	k-means clustering
classification	interestingness association	database design	multi-label classification	aggregate answering
text classification	hierarchical text classification	database research	semi-supervised classification learning	features genre classification
active learning	active learning text	pac learning	statistical machine learning	subspace clustering

Figure 5.2: Examples of user study questions

	Correct Number	Percentage
User 1	27	90%
User 2	22	73.3%
User 3	24	80%
User 4	24	80%

Figure 5.3: Unrelated topic identification

CHAPTER 6

CONCLUSION

In this thesis, we have proposed a novel probabilistic approach to build a topic map by using paradigmatic and syntagmatic relations from large text data based on random walk defined on adjacency graph. This approach is completely unsupervised and can be generalized to any kind of text data. Our model is not restricted by the characteristics of languages. Therefore, our approach can be applied to multi-lingual text data. The topic map constructed in this thesis meets user requirements 1, 2 and 4. Evaluation results show that this approach is effective and can generate very meaningful topic map. One limitation of the work is that we only experimented with one collection; additional evaluation of the proposed method on additional test set is obviously needed to further confirm the effectiveness of the proposed method. Another interesting future direction is to integrate the map generated using our algorithms with an existing search engine so as to support integrated browsing and querying.

REFERENCES

[1] S. E. Robertson, "Readings in information retrieval," K. Sparck Jones and P. Willett, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, ch. The Probability Ranking Principle in IR, pp. 281–286. [Online]. Available: http://dl.acm.org/citation.cfm?id=275537.275701

[2] N. Fuhr, "A probability ranking principle for interactive information retrieval," *Inf. Retr.*, vol. 11, no. 3, pp. 251–265, June 2008. [Online]. Available: http://dx.doi.org/10.1007/s10791-008-9045-0

[3] A. J. Zhou, J. Luo, and H. Yang, "DUMPLING: A novel dynamic search engine," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, 2015. [Online]. Available: http://doi.acm.org/10.1145/2766462.2767873 pp. 1049–1050.

[4] D. Guan, S. Zhang, and H. Yang, "Utilizing query change for session search," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '13. New York, NY, USA: ACM, 2013. [Online]. Available: http://doi.acm.org/10.1145/2484028.2484055 pp. 453–462.

[5] Y. Zhang and C. Zhai, "Information retrieval as card playing: A formal model for optimizing interactive retrieval interface," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, 2015. [Online]. Available: http://doi.acm.org/10.1145/2766462.2767761 pp. 685–694.

[6] Y. Zhang and C. Zhai, "A sequential decision formulation of the interface card model for interactive IR," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, July 17-21, 2016*, 2016. [Online]. Available: http://doi.acm.org/10.1145/2911451.2911543 pp. 85–94.

[7] O. Maimon and L. Rokach, *Data Mining and Knowledge Discovery Handbook*, 2nd ed. Springer Publishing Company, Incorporated, 2010, ch. Clustering methods.

[8] R. Sibson, "Slink: An optimally efficient algorithm for the single-link cluster method," *The Computer Journal*, vol. 16, no. 1, pp. 30–34, 1973. [Online]. Available: + http://dx.doi.org/10.1093/comjnl/16.1.30

[9] C. C. Aggarwal and C. Zhai, "A survey of text clustering algorithms." in *Mining Text Data*, C. C. Aggarwal and C. Zhai, Eds. Springer, 2012, pp. 77–128. [Online]. Available: http://dblp.uni-trier.de/db/books/collections/Mining2012.htmlAggarwalZ12a

[10] N. Oikonomakou and M. Vazirgiannis, "A review of web document clustering approaches," in *Data Mining and Knowledge Discovery Handbook*. Springer, 2010, pp. 931–948.

[11] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *In KDD Workshop on Text Mining*, 2000.

[12] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey, "Scatter/gather: A cluster-based approach to browsing large document collections," in *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '92. New York, NY, USA: ACM, 1992. [Online]. Available: http://doi.acm.org/10.1145/133160.133214 pp. 318–329.

[13] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, no. 4, pp. 77–84, Apr. 2012. [Online]. Available: http://doi.acm.org/10.1145/2133806.2133826

[14] D. M. Blei, "Introduction to probabilistic topic models," in *In Communications of the ACM*, 2011.

[15] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '99. New York, NY, USA: ACM, 1999. [Online]. Available: http://doi.acm.org/10.1145/312624.312649 pp. 50–57.

[16] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[17] H. M. Wallach, "Topic modeling: Beyond bag-of-words," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1143844.1143967 pp. 977–984.

[18] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *Advances in Neural Information Processing Systems 20*, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds. Curran Associates, Inc., 2008, pp. 121–128. [Online]. Available: http://papers.nips.cc/paper/3328-supervised-topic-models.pdf

[19] X. Wang, A. McCallum, and X. Wei, "Topical n-grams: Phrase and topic discovery, with an application to information retrieval," in *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, ser. ICDM '07. Washington, DC, USA: IEEE Computer Society, 2007. [Online]. Available: http://dx.doi.org/10.1109/ICDM.2007.86 pp. 697–702.

[20] Z. Liu, W. Huang, Y. Zheng, and M. Sun, "Automatic keyphrase extraction via topic decomposition," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1870658.1870694 pp. 366–376.

[21] W. X. Zhao, J. Jiang, J. He, Y. Song, P. Achananuparp, E.-P. Lim, and X. Li, "Topical keyphrase extraction from twitter," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=2002472.2002521 pp. 379–388.

[22] X. Ren, Y. Lv, K. Wang, and J. Han, "Comparative document analysis for large text corpora," *CoRR*, vol. abs/1510.07197, 2015. [Online]. Available: http://arxiv.org/abs/1510.07197

[23] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=944919.944937

[24] D. Mimno, W. Li, and A. McCallum, "Mixtures of hierarchical topics with pachinko allocation," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 633–640.

[25] D. M. Blei, T. L. Griffiths, and M. I. Jordan, "The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies," *Journal of the ACM (JACM)*, vol. 57, no. 2, p. 7, 2010.

[26] X. Wang, B. Tan, A. Shakery, and C. Zhai, "Beyond hyperlinks: organizing information footprints in search logs to support effective browsing," in *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM, 2009, pp. 1237–1246.

[27] C. Wang, M. Danilevsky, N. Desai, Y. Zhang, P. Nguyen, T. Taula, and J. Han, "A phrase mining framework for recursive construction of a topical hierarchy," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '13. New York, NY, USA: ACM, 2013. [Online]. Available: http://doi.acm.org/10.1145/2487575.2487631 pp. 437–445.

[28] K. Ahmed and G. Moore, "An introduction to topic maps," *IBM Systems Journal,*, 2005.

[29] K. Rajkumar, "Topic map: An ontology framework for information retrieval," *CoRR*, vol. abs/1003.3530, 2010. [Online]. Available: http://arxiv.org/abs/1003.3530

[30] S. Jiang and C. Zhai, "Random walks on adjacency graphs for mining lexical relations from big text data," in *Big Data (Big Data), 2014 IEEE International Conference on*. IEEE, 2014, pp. 549–554.

[31] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-graber, and D. M. Blei, "Reading tea leaves: How humans interpret topic models," in *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, Eds., 2009, pp. 288–296. [Online]. Available: http://books.nips.cc/papers/files/nips22/NIPS2009$_0$125.*pdf*