PatentDom: Analyzing Patent Relationships on Multi-View Patent Graphs

Longhui Zhang[†] Lei Li[†] Tao Li[†] Dingding Wang[‡]

[†]School of Computing and Information Sciences
Florida International University
11200 S.W. 8th Street
Miami, FL 33199
{lzhan015,lli003,taoli}@cs.fiu.edu

[‡]Department of Computer & Electrical Engineering and Computer Science
Florida Atlantic University
777 Glades Road
Boca Raton, FL 33431
wangd@fau.edu

ABSTRACT

The fast growth of technologies has driven the advancement of our society. It is often necessary to quickly grasp the linkage between different technologies in order to better understand the technical trend. The availability of huge volumes of granted patent documents provides a reasonable basis for analyzing the relationships between technologies. In this paper, we propose a unified framework, named `PatentDom`, to identify important patents related to key techniques from a large number of patent documents. The framework integrates different types of patent information, including patent content, citations of patents, and temporal relations, and provides a concise yet comprehensive technology summary. The identified key patents enable a variety of patent-related analytical applications, e.g., outlining the technology evolution of a particular domain, tracing a given technique to prior technologies, and mining the technical connection of two given patent documents. Empirical analysis and extensive case studies on a collection of US patent documents demonstrate the efficacy of our proposed framework.

Categories and Subject Descriptors: H.3.3[Information Storage and Retrieval]: Information Search and Retrieval

Keywords: Patent Analysis; Patent Evolution; Dominating Set; Steiner Tree; Center-Piece Subgraph

1. INTRODUCTION

Technological innovation is becoming one of the important factors that stimulate the development of our society. Granted patents, as the major carrier for technology documentation, have great potential to provide valuable insights of technologies. Analyzing patent documents enables us to effectively understand technological progress, comprehend the evolution of technologies and capture the emergence of new technologies [3, 6, 29].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CIKM'14, November 3–7, 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2662031.

One representative application of patent analysis involves that enterprises evaluate the prior art or technology evolution of a specific technical field in the development of new products. To conduct such an analysis, a key step is to identify important patents from a large number of related patent documents, where these patents can represent dominating technologies in the corresponding technical field [18]. In addition, for a technology company who maintains a large number of patents, it is often time-consuming and costly to manually examine these patents to identify the important ones for further maintenance. Automatic discovery of key patents from patent collections is able to help improve the efficiency and reduce the cost of patent portfolio management. Further, connecting the dots between the identified key patents enables a variety of patent analysis tasks.

In this paper, we study the problem of mining dominating technologies from a large collection of patent documents. Previous research efforts [8, 11, 21] tackle this problem via clustering or topic-based mining, where the key patents are essentially identified through content analysis. However, as a scientific means of technology documentation with legal significance and potential economic values, a patent document often has complex structures and special terminologies. The sophisticated patent language poses great challenges to automatic patent analysis, and hence it is difficult to identify key patents purely based on patent content.

In the domain of patent analysis, patent documents are often explicitly organized using citation links [9]. The citation relations among patents documents provide good indicators for the importance of patents. Representative work involves [26, 27], which utilizes the co-citation relations of patent documents to identify key patents. However, citations among patent documents are usually sparse, which may result in the technology gap, and consequently hinder the comprehension of dominating technologies.

To address the aforementioned issues, in our work, we explore the possibility of integrating both patent content and citation relations in identifying key patents. To this end, we propose a unified framework, named `PatentDom`, in which multiple types of patent-related information are employed, including the content and citation relations of patent documents. The input to the system is a topic or a classification code relevant to a specific technical field. The system first retrieves all the patent documents related to the topic/code from a patent database. We then construct a multi-view patent graph in which patent content, citation relations and

temporal orders are integrated. We model the problem of identifying key patents as a minimum-cost dominating set problem, and select key patents using an approximation algorithm. We further discover a list of patent-related problems based on the identified key patents. These problems can be resolved by considering the temporal order of patent documents and connecting the dots between the key patents through graph-based algorithms.

To the best of our knowledge, our work is the first journey towards unifying the process of understanding the linkage between different technologies in the domain of patent analysis, by considering both document content and citation relations of patents. In summary, the contributions of our work are three-fold:

- We present a unified framework to identify dominating technologies on a multi-view patent graph that synthesizes both patent content and citation relations.
- We apply the proposed framework to multiple patent-related analysis problems that aim to discover the linkage of patents, including:
  - `PatentLine`, i.e., to outline the technology evolution of a particular domain;
  - `PatentTrace`, i.e., to trace a given technique to previous related technologies;
  - `PatentLink`, i.e., to discover the technical connection of two given patent documents.
- We conduct extensive empirical evaluation on a collection of US patent documents, and the results demonstrate the efficacy of the framework.

The rest of the paper is organized as follows. Section 2 presents a brief summary of prior work relevant to key patent discovery. In Section 3, we formalize the problem and describe the algorithmic details of our proposed framework. We then present several potential patent-related applications and the corresponding solutions in Section 4. Empirical evaluation of our framework is reported in Section 5. Finally Section 6 concludes the paper.

2. RELATED WORK

Automatic key patent discovery is an emerging problem in the domain of patent analysis. In the past decades, a couple of interesting methods have been proposed to address this problem. Generally speaking, it is tackled by researchers from either content analysis or citation analysis. In the following, we highlight the previous research that are most relevant to our work.

A patent document is often lengthy with rich content, consisting of descriptions, embodiments, claims, etc. These unstructured texts aim to depict the background of the invention, and describe the scope of protection conveyed by the invention. Based on the rich content, researchers identify important patents by exploring the novelty or influence of patent documents. For example, Shaparenko et al. [21] assume that a document is novel and influential if it has fewer similar documents published before it, and has more similar documents published after it. Hasan et al. [8] analyze patent novelty from patent claims by considering the time difference between a keyword's first appearance in patents and the issue year of the subject patent.

However, patents are often full of technical terms and ambiguous expressions. It is quite difficult to quantify the similarity of patent documents or discover the identity of patent keywords without the help of external resources. To alleviate this issue, Hu et al. [11] propose a topic-based mining approach to quantify a patent's novelty and influence, and report promising results of recognizing core patents. Nonetheless, their method is restricted by the latent property of topics, and hence it is difficult to provide appropriate explanation on the novelty/influence of patents.

Another direction of identifying key patents is to analyze the citation relations of patent documents. Representative work involves [26, 27], which utilizes the co-citation relations of patent documents to identify key patents. The method employs citation frequencies and the ages of patents to avoid overemphasizing "older" patents. However, citations among patent documents are usually sparse, and sometimes it may result in the technology gap, which will hinder the comprehension of dominating technologies in a technical field. Even if the citation links are not sparse, the identified key patents may belong to the same company/inventor, and hence the coverage of key patents is sacrificed.

Our work is different from the aforementioned approaches, as we consider both patent content and citation relations when identifying key patents. In addition, we model the problem of discovering key patents as a minimum-cost dominating set problem. The property of "minimum dominating set" provides the maximum coverage [13] of technologies in a technical field. In recent years, this concept has been applied to other domains of information retrieval, e.g., document summarization [22] and storyline generation [16, 25], aiming to find dominating sentences from a large pool of documents. However, to the best of our knowledge, there is no research effort that applies the concept of dominating set to the patent domain. In additions, the identified three applications have real needs in patent analysis.

3. IDENTIFYING DOMINATING PATENTS

In the domain of patent analysis, it makes more sense to restrict the scope to a particular technical field. Hence, given a classification code related to a specific technical field, we initially retrieve all available patent documents under the code from a patent database. The problem of identifying key patents can be defined as follows:

PROBLEM 1. *Given a collection of granted patents $D = \{d_1, d_2, \ldots, d_n\}$, extract a subset of patents $P \subseteq D$, where $P = \{p_1, p_2, \ldots, p_m\}$ and each p_i denotes a key patent that can represent the dominating technology within the patent collection.*

PROBLEM 1 gives us a generic definition of key patents, which can be used to describe the general problems of key patent discovery. In some cases, patent analysts expect to obtain important patents with respect to specific queries, e.g., a set of query patents. Then PROBLEM 1 can be redefined as follows:

PROBLEM 2. *Given a collection of granted patents $D = \{d_1, d_2, \ldots, d_n\}$ and a set of query patents $Q = \{q_1, q_2, \ldots, q_k\}$, extract a subset of patents $P \subseteq D$, where $P = \{p_1, p_2, \ldots, p_m\}$ and each patent p_i is able to represent the dominating technology related to the query set Q.*

To address the aforementioned problems, we propose a unified framework, named `PatentDom`, which employs the minimum dominating set of a patent graph to represent the key patents. Specifically, we first construct a multi-view patent graph using the information of patent content, citation relations and temporal orders of patent documents, and then identify dominating/influential patents from the graph. Taking PROBLEM 1 as an example, we can assume that the extracted key patents should represent all the patent documents (i.e., every patent in the collection should be relevant to the extracted patents in terms of technologies). In other words, these key patents serve as a brief summary of the entire patent collection. Meanwhile, the number of these patents should be as small as possible. Such a summary of the patent collection under the above assumption is exactly the minimum dominating set of the patent graph. We hence model the problems as a minimum-cost dominating set problem, where the cost can be defined using different types of information, depending on the problem being solved. The framework is described in Figure 1.

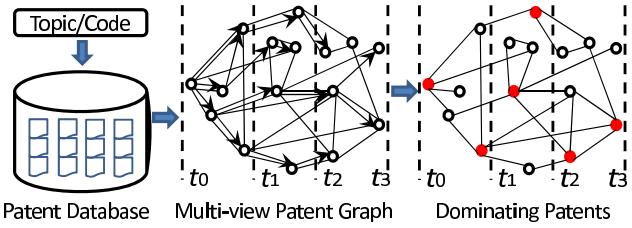

Figure 1: An overview of the framework.

3.1 Constructing Multi-View Patent Graph

As introduced in Section 1, the patent data consists of multiple types of information that shape the relations among patent documents. We use a multi-view graph $\mathbb{G}$ to represent these relations, where $\mathbb{G} = (V, \mathbf{w}_v, E_s, \mathbf{w}_s, E_{ct}, \mathbf{w}_{ct})$.

$\mathbb{G}$ contains a set of nodes/vertices (patent documents) V, where each node $v \in V$ is associated with a cost value w_v and a timestamp t. In our problem setting, the cost w_v can be defined using the information of patent content and/or citation relations. For example, to address PROBLEM 1, the cost can be calculated as the inverse of the total number of citations of the corresponding patent document, as we expect the selected patent is more influential than others. When selecting dominating nodes, the total cost of selected nodes should be minimized.

In addition, the vertices are connected by two types of edges: E_s and E_{ct}. E_s contains undirected edges, where each edge connects two patent vertices and the edge weight w_s denotes the content proximity of connected vertices. For patent documents, it is often difficult to calculate the similarity/proximity, as there are a lot of domain-specific and ambiguous terms, and different patents may have their own writing styles. To this end, we extract the most significant section of patents, i.e., `claims`, since this section defines the major invention of patents and often has relatively stable writing structures. We employ "bag-of-words" representation and the cosine measure for proximity computation. Two vertices are linked if and only if the content proximity is greater than a predefined threshold δ. In our proposed framework, E_s is used for dominating patent selection.

Another set of edges, E_{ct}, are directed edges, where each edge represents either the citation linkage between two vertices, or the temporal order of two vertices. Two vertices form a temporal link if and only if they do not have a citation link and their respective timestamp difference falls into a predefined time range $[\tau_1, \tau_2]$. For simplicity, we assign a unit value 1 to the weight of edges E_{ct}, i.e., $w_{ct} = 1$. E_{ct} serves to connect the selected dominating patents for specific patent applications. For example, to outline the technology evolution of a particular technical field, we can employ E_{ct} to generate an evolution tree of dominating patents. Details can be found in Section 4 for different applications.

3.2 Identifying Dominating/Influential Patents

Our goal is to detect the patent documents with representative power, or say, dominating/influential patents. To this end, we define the problem on the undirected part, i.e., $(V, \mathbf{w}_v, E_s, \mathbf{w}_s)$, of the multi-view graph introduced in Section 3.1. Specifically, given the graph $\mathbb{G}$, a *dominating* set of $\mathbb{G}$ is a subset S of vertices with the following property: each vertex $v \in V$ is either in the dominating set S, or is adjacent to some vertices in S. Note that in $\mathbb{G}$, each vertex has a cost with respect to specific applications. The problem of finding a set of dominating patent documents can be formulated as the minimum-cost dominating set problem [5, 22].

PROBLEM 3. *Given a graph $\mathbb{G} = (V, \mathbf{w}_v, E_s, \mathbf{w}_s)$ and a budget L, the problem of minimum-cost dominating set (MCDS) is to find a dominating set S, with size L, of vertices in $\mathbb{G}$ whose total vertex cost is the minimum.*

The MCDS problem is closely related to the problem of minimum dominating set (MDS). The vertex cover problem, which is known as an NP-hard problem, can be reduced to the MDS problem.

REDUCTION. Given a connected graph $\mathbb{G} = (V, E)$, we replace each edge of $\mathbb{G}$ by a triangle to create another graph $\mathbb{G}' = (V', E')$. In $\mathbb{G}'$, $V' = V \cup V_e$ where $V_e = \{v_{e_i} | e_i \in E\}$, and $E' = E \cup E_e$ where $E_e = \{(v_{e_i}, v_k), (v_{e_i}, v_l) | e_i = (v_k, v_l) \in E\}$. Such a transformation can be viewed as subdividing each edge (u, v) by the addition of a vertex, and adding an edge directly from u to v.

Assume $\mathbb{G}$ has a vertex cover S with size K, then S forms a dominating set in $\mathbb{G}'$. As each vertex v has at least one edge (v, u), and u must be in the cover if v is not. Since v is adjacent to u, then v has a neighbor in S.

For the reverse direction, assume that $\mathbb{G}'$ has a dominating set S' with size K, which only contains vertices from the vertex set V. If v_{e_i} is selected in S', then we can replace it by either v_k or v_l, without increasing the size of S'. We now claim that S' forms a vertex cover. For each edge e_i, v_{e_i} must have a neighbor (either v_k or v_l) in S'. This neighbor will cover the edge e_i, and thus the dominating set in $\mathbb{G}'$ is a vertex cover in $\mathbb{G}$. □

It has been shown that no algorithm can achieve an approximation factor better than $c \log |V|$ for some $c > 0$ [13]. However, we can obtain a greedy approximation for MCDS, as shown in Algorithm 1. Starting from an empty set, if the current subset of vertices is not the dominating set, a new vertex with the minimum averaged cost (with respect to its neighbor size) and not adjacent to any vertex in the current set will be added. In other words, the cost of the new vertex can be evenly shared by its neighbors. Such a greedy

algorithm provides a factor of $1 + \log |V|$ approximation of MCDS [20].

Algorithm 1: Approximation of MCDS.

Input: $\mathbb{G} = (V, \mathbf{w}_v, E_s, \mathbf{w}_s)$: undirected patent graph
L: predefined threshold of dominating patents
Output: minimum-cost dominating set S

$S \leftarrow \emptyset$; $T \leftarrow \emptyset$
while $|S| < L$ **do**
  for $v \in V - S$ **do**
    $s(v) = |\{v' | (v', v) \in E_s\} \setminus T|$
  $v^* = \arg\min_v \frac{cost(v)}{s(v)}$
  $S = S \cup \{v^*\}$; $T = T \cup \{v' | (v', v^*) \in E_s\}$
return S

By Algorithm 1, we can obtain a set of dominating patents related to the specific technical field, with the limit of a predefined dominator number L. Note that in Algorithm 1, $cost(v)$ represents the value of $w(v)$, i.e., the cost of the vertex v. It may be related to the citation relations as indicated in Problem 1, or relevant to the query set as indicated in Problem 2.

4. POTENTIAL APPLICATIONS

The identified dominating patents from Section 3.2 enable a list of patent-related applications. In this section, we will discuss these applications from the perspective of connecting the dots between dominating patents.

4.1 Generating Tree-Based PatentLine

The first application is named as `PatentLine`, aiming to discover the technology evolution tree of a particular technical field. This problem has recently attracted increasing interest in the information retrieval community. Most existing approaches focus on identifying evolutionary topics in scientific literatures [1, 2] by making use of vector space model or LDA-alike topic models. Some recent work further tries to analyze the roles of linkage analysis (e.g., the co-authorship [30] or citation analysis [9]) in topic detection and evolution. However, these existing methods cannot be simply applied to our problem setting of generating an evolutionary tree of patents. In addition, the characteristics of patent domain (e.g., lengthy and ambiguous description, full of technical terms) render these methods ineffective in generating patent evolution tree.

The dominating patents obtained from dominating set approximation are capable of representing the rest of patents in the graph in terms of content proximity and citation influence. Note that when utilizing Algorithm 1 to identify dominating patents, the cost of a vertex, i.e., $cost(v)$, is defined as the inverse of the total number of citations of the corresponding patent document, as we expect the selected patent is more influential than others. However, there might be some technical gaps among these patents, that is, they may not be well connected. In order to provide a fluent structure of patent documents, e.g., a patentline, we have to find ways to link them together. Also, for presentation purpose, the generated structure of patent documents should be as dense and informative as possible, i.e., to include the minimum number of patents or have the maximum influence over other options.

Based on our previous work [28], we utilize the directed part, i.e., $(V, \mathbf{w}_v, E_{ct}, \mathbf{w}_{ct})$, of the multi-view graph introduced in Section 3.1. The procedure is depicted in Figure 2.

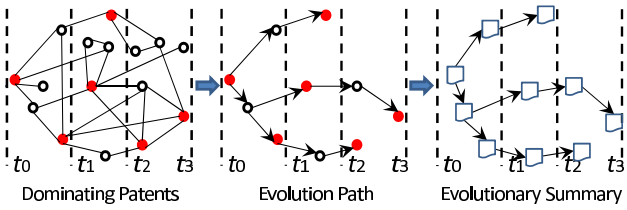

Figure 2: The procedure of `PatentLine`.

We formulate the problem as the minimum-cost Steiner tree problem. Given a graph $\mathbb{G}$ and a subset of vertices S, a Steiner tree of $\mathbb{G}$ is similar to minimum spanning tree, defined as the subtree of $\mathbb{G}$ that contains S with the minimum total cost. In our problem setting, the total cost is defined as the sum of vertex cost of the entire Steiner tree.

Problem 4. *Given a graph $\mathbb{G} = (V, \mathbf{w}_v, E_{ct}, \mathbf{w}_{ct})$, a vertex set $S \subset V$ (terminals) and a vertex $v_0 \in S$ from which every vertex of S is reachable in $\mathbb{G}$, the problem of minimum-cost Steiner tree (MCST) is to find the subtree of $\mathbb{G}$ rooted at v_0 that subsumes S with minimum total vertex cost.*

Algorithm 2: $Steiner_i(\mathbb{G}, S, v_0, k)$

Input: $\mathbb{G} = (V, \mathbf{w}_v, E_{ct}, \mathbf{w}_{ct})$: directed patent graph
S: terminal set
$v_0 \in S$: root of the Steiner tree
k: target size of terminals to be covered
Output: **T**: a Steiner tree rooted at r_0 covering at least k terminals

$T \leftarrow \emptyset$
while $k > 0$ **do**
  $T_{opt} \leftarrow \emptyset$; $cost(T_{opt}) \leftarrow \infty$
  for v, $(v_0, v) \in E_{ct}$, *and* k', $1 \le k' \le k$ **do**
    $T' \leftarrow Steiner_{i-1}(\mathbb{G}, S, v, k') \cup \{(v_0, v)\}$
    if $(cost(T_{opt}) > cost(T'))$ **then**
      $T_{opt} \leftarrow T'$
  $T \leftarrow T \cup T_{opt}$; $k \leftarrow k - |S \cap V(T_{opt})|$;
  $S \leftarrow S \setminus V(T_{opt})$
return T

The problem of MCST, a directed version of the Steiner tree problem, is known as an NP-hard problem [14]. As suggested by [4], a reasonable approximation can be achieved by finding the shortest path from the root to each terminal and then combining the paths, with the approximation ratio of $O(\log^2 k)$, where k is the number of terminals. The approximation algorithm is described in Algorithm 2.

The algorithm employs a recursive way to generate the Steiner tree T. It takes a level parameter $i \ge 1$. When $i = 1$, $Steiner_1$ is simple to describe, i.e., to find the k terminals which are the closest to the root v_0 and connect them to v_0 using shortest paths. As $i > 1$, $Steiner_i$ repeatedly finds a vertex v adjacent to the input root of the i-th function and a number k' such that the cost of the updated tree is the

least among all the trees of this form. After obtaining the expected path, we update the corresponding Steiner tree, the target size k and the terminal set S.

The generated Steiner tree of the patent graph gives us an elegant representation of patent evolution, which describes the transitions from the root patent to all the other dominating patents. Once the Steiner tree is generated, we can easily obtain a concise summary for each patent in the tree by applying document summarization techniques [15].

4.2 Tracing Technologies To Ancestors

The second application is called `PatentTrace`, which aims to trace a given patent document back to its ancestors to examine what techniques that the given patent utilizes. This problem is relatively new in the domain of patent analysis. One major issue of modern patent documents is the growing complexity of the involved tasks, i.e., a single patent may contain a list of procedures and involve a lot of technologies. For such inventions, one may often need multiple research teams to develop different processes, and various inventions may be interlinked. Hence, to ease the understanding of patent analysts, it is imperative to identify key techniques related to the patent being investigated, and represent them in an informative manner.

To tackle this problem, we rely on the identified dominating patents based on the framework of `PatentDom`. The procedure of `PatentTrace` is described in Figure 3.

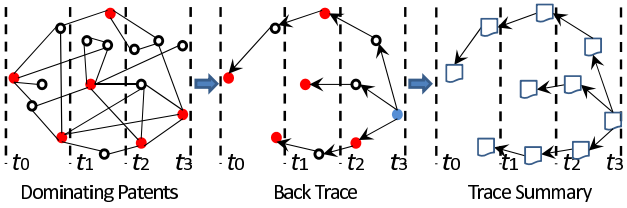

Figure 3: The procedure of `PatentTrace`.

Given a patent document as a query q, we first utilize Algorithm 1 to discover dominating patents based on the undirected part of the multi-view patent graph introduced in Section 3.1. Here we expect that the dominating patents are not only relevant to the query patent, but also reflect the important technologies. Hence in Algorithm 1, the cost of a vertex, i.e., $cost(v)$, should be defined in a way different from the one introduced in Section 4.1. To this end, we consider both content and citation relations of patent documents, and define $cost(v)$ as

$$cost(v) = \frac{1 - sim(v, q)}{citation(v)}, \tag{1}$$

where the numerator denotes the content distance between the query patent q and the node v, and the denominator represents the citation count of the patent v. The similarity between patents is calculated using the content from `claims`, as indicated in Section 3.1. By Eq.(1), we expect to select the patents with content similar to the query patent, as well as with more citations to represent its influential power.

After identifying a list of dominating patents related to the given query, the next step is to connect these patents in order to provide a fluent trace from the query back to its ancestors. Some of the identified key patents may have a timestamp later than the one of the query patent, and hence they cannot be included in the final trace. To this end, we employ the directed part of the multi-view patent graph. Starting from the query node, we iteratively reverse the directed edges, and remove the nodes later than the query node, as well as the edges with opposite directions. The resulted subgraph $\mathbb{G}^*$ serves as the basis for trace generation.

Similar to `PatentLine`, we formulate the problem of tracing a patent to its ancestors as the minimum-cost Steiner tree problem. We then utilize Algorithm 2 to form the trace. The input is slightly different from the one in Section 4.1. v_0, as the root of the Steiner tree, is the query patent q. The terminal set S contains the dominators that are reachable from v_0 in the subgraph $\mathbb{G}^*$. The generated Steiner tree presents an informative representation of patent trace, which vividly describes the related ancestor technologies with respect to the query patent. Similar to `PatentLine`, we can generate a concise summary for each patent in the tree by applying document summarization techniques.

4.3 Discovering Technical Connections

The third application is named as `PatentLink`, aiming at discovering the potential relations between two patent documents. Given two patents p_1 and p_2 from different time periods, where p_1 is published earlier than p_2, they may not connect directly through citation relations. However, it is possible that p_2 is an implicit extension of p_1 in terms of technologies, or an application of the techniques described in p_1. Such latent connections are valuable for companies to design the corresponding product strategy. To the best of our knowledge, this problem has not yet attracted any research attention in the domain of patent analysis.

To address this problem, we first utilize the framework of `PatentDom` to identify dominating patents. Given a query set $Q = \{p_1, p_2\}$, we discover the dominating patents relevant to Q using Algorithm 1. The calculation of vertex cost is similar to Eq.(1). The only difference is the similarity score, which is computed as the averaged similarity between the vertex and the query patents.

The key patents are able to help connect the two query patents. However in the multi-view patent graph, multiple paths may exist between the given query patents. The challenge here is how to identify important paths in order to depict the strong connection between queries. In other words, how to find the nodes that are the center-piece, and have direct or indirect connections to all the query nodes? To this end, we employ the so-called center-piece subgraph [23, 24] and apply it to the direct part of the multi-view graph. We expand the query set Q by adding all the dominators falling in between the time period from p_1 and p_2. By doing this, the generated center-piece subgraph is able to show how the two patents are connected through leading technologies. The procedure is shown in Figure 4.

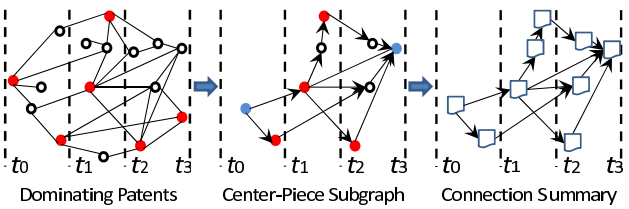

Figure 4: The procedure of `PatentLink`.

The algorithm CEPS described in [24] for generating centerpiece subgraph involves three steps: (1) calculating individual goodness score for a single node with respect to each query node; (2) combining individual scores to obtain the goodness score for a single node with respect to the query set; and (3) extracting a connection subgraph maximizing the goodness criteria. The individual goodness score can be calculated using random walk with restart. Given a query p_i, a random particle starts from p_i, and then iteratively transmits to its neighborhood with the probability proportional to the edge weight between them, and also at each step, it has some probability c to return to the node p_i. Let $\mathbf{R}$ be the matrix containing the probability that the particle will finally stay at node p_i, then the matrix form of random walk calculation can be represented as

$$\mathbf{R}^T = c\mathbf{R}^T \times \mathbf{W} + (1-c)\mathbf{E},$$

where $\mathbf{E} = [\vec{e_i}](i = 1, \ldots, |Q|)$, and each $\vec{e_i}$ is the unit query vector with all zeros except one at row p_i. Notice that in our problem setting, we expand the query set by including appropriate dominators, and hence the corresponding dominators' entries are 1. $\mathbf{W}$ is the normalized adjacency matrix. Detailed procedure can be found in [24].

5. EMPIRICAL EVALUATION

In this section, we provide a comprehensive experimental evaluation to show the efficacy of our proposed framework PatentDom. We start with an introduction to the patent collection used in the experiment. To validate the proposed framework, we compare our method with other existing solutions of identifying key patents. We further present several case studies to show the efficacy of the approaches for different applications.

5.1 Patent Data

The data set used in our experiment is obtained from the State Intellectual Property Office of the P.R.C (SIPO)[1], containing 16,518 US granted patents under the section G (physics), whose filing dates are ranging from 2001 to 2012. It covers three sub-domains, including patents related to data processing system (G06Q 10/00), photomechanical production (G03F 7/00), and optical operation (G02F 1/00). The statistics of the data are depicted in Table 1. Under each patent code, there are a list of major patent groups, and each group contains at least 250 patents. Note that there is no standard patent data set that provides the ground truth of important patent documents with respect to a domain. Hence, for evaluation purpose, we ask patent analysts to manually select at least 20 key patents for each patent group as the ground truth.

Table 1: The description of patent data.

Domain Code	Groups	# of Patents	Average
G02F 1/00	17	11,218	660
G03F 7/00	6	2,922	487
G06Q 10/00	5	2,378	476

To conduct the experiment, we extract the title, claims, citations and publishing timestamp of each patent document, and preprocess the content using natural language processing techniques, such as removing stop words, tokenizing, and stemming. The content of each patent is represented as a term vector, and the content proximity of patents is calculated using the cosine similarity for the purpose of similarity calculation. The citation relations are restricted in the patent collection.

[1]http://english.sipo.gov.cn.

5.2 Evaluation on PatentDom

In PatentDom, to construct the multi-view patent graph, we empirically set the content proximity threshold δ as 0.2, and the time range as 3 months. To evaluate our proposed framework, we implement three existing methods of identifying key patents as the baselines:

- COA [8]: It rates a patent based on its value by measuring the recency and impact of important phrases that appear in the claims. The score of a word w in a patent d is determined as follows:

  $$score(w) = \max(\frac{support(w) - 2}{age(w) + 1}, 0),$$

  where $age(w)$ defines the recency of w, which is the time difference between the year w first occurs in the patent collection and the issue year of d; $support(w)$ is the number of follow-up patents that contain w. The score of d is the sum of scores of all the words in d. This method is based on the content and temporal information of patent documents.

- PageRank [7]: It employs PageRank to rank patent documents, where the probability of accessing a patent is treated as the citation-based score for each document. This method is purely based on the citation relations of patents.

- CorePatent [11]: It aims to address the unique patent vocabulary usage problem by using a topic-based temporal mining approach to quantify a patent's novelty and influence. It initially identifies latent topics using an LDA-alike model [19], and then examines the activeness of topics and removes noisy topics. Finally it quantifies patent novelty and influence, and ranks patents by their scores. This method utilizes both content and temporal information of patents.

The problem of identifying key patents is essentially a retrieval problem. For each method and each patent group, we rank and select top@10, top@30, top@50 patent documents based on its corresponding ranking criterion, and compare the results with the ground truth provided by patent analysts. For comparison, we compute the averaged precision, recall and F1-score of the entire 28 patent groups. The results are reported in Table 2.

As depicted in the table, our proposed framework, PatentDom, achieves the best performance compared with other baselines in terms of the precision, recall and F1-score. Especially for the recall, it significantly outperforms other methods. This is very valuable as the retrieval in the domain of patent analysis is a recall-based task. It is extremely important to have a higher recall in order to reduce the human efforts as well as to lower the risk of missing important patent documents.

We further examine the details of the results by investigating the content as well as citations of patent documents. Based on the analysis, we observe that PatentDom presents important patents of different time periods, and

Table 2: Comparison with existing methods. (Bold indicates the best performance. * indicates the statistical significance at $p < 0.01$.)

Methods	top@10			top@30			top@50		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
COA	0.11	0.056	0.07	0.092	0.138	0.11	0.086	0.215	0.123
PageRank	0.106	0.053	0.07	0.1	0.15	0.12	0.112	0.28	0.16
CorePatent	0.188	0.094	0.125	0.192	0.288	0.231	0.192	0.48	0.274
PatentDom	**0.194**	**0.097**	**0.129**	**0.22***	**0.33***	**0.263***	**0.212***	**0.53***	**0.3***

these patents are able to cover the dominating technologies in the corresponding domain without too much interlinking. Compared with PatentDom, the baselines provide partial or unreasonable key patents:

- Most patents in the results of COA fall into the earlier time periods, i.e., it only identifies key patents in the early years. However, there might be some patents serving as a connection link between the preceding and the following technologies, which are also important. COA fails to capture these patents, and hence its performance is comparatively worse than other baselines.

- PageRank only identifies important patents in the early and middle stages, due to the property of the PageRank algorithm. However in practice, technologies often evolve over time, and hence in recent stages we may have emerging technologies used by a lot of companies, which are also important in some sense.

- CorePatent discovers important patents from the topic-oriented perspective, and the results generated by this method are important in terms of the content. However, it fails to consider the citation relations of patent documents. Because of this, the identified key patents often center on several major technology companies, e.g., FujiFilm Corporation presents a lot of patents in photomechanical production. Nonetheless, these patents are usually related to each other with much more redundancy. This is the reason for which the performance of CorePatent is comparable to ours when the number of retrieved key patents is small, but is getting worse with more key patents.

5.3 Case Studies of Different Applications

Validating the efficacy of our proposed solutions to the three applications is a subjective process, as it is difficult to obtain annotated ground truth. We hence resort to case studies on the collected patent data. All the cases used in this section are reviewed by domain experts and are confirmed to be effective.

5.3.1 A Case Study on PatentLine

PatentLine presents a way to explore the technology evolution of a specific technical field. To evaluate the efficacy of PatentLine, we perform a case study on a collection of patent data. The major international classification code of the patent data is "G06Q 10/00", representing the topic of "data processing systems or processes for administration and management of an organization, enterprise or employees". This code includes 5 sub-domains, and their descriptions are shown in Table 3.

Table 3: The description of patent classification.

Code	Description	# of Patents
G06Q 10/02	Reservations, e.g., meetings	288
G06Q 10/04	Forecasting or optimization	341
G06Q 10/06	Workflow management	404
G06Q 10/08	Inventory management	534
G06Q 10/10	Office automation	811

We run Algorithm 1 (limiting the number of dominators to be 10) and Algorithm 2 on the generated multi-view patent graph, and the resulted Steiner tree is demonstrated in Figure 5, organized by the temporal order of patents. For representation purpose, we only list the keywords that are contained in the title of patents. The **bold** rectangles denote the dominators identified by Algorithm 1. The X-axis describes the publishing dates of the patents. As observed in Figure 5, "Management" in "G06Q 10/00" starts from manipulating data, as described in the first dominator, and then can be decomposed into several subtopics. The line labeled as ① mainly describes meeting scheduling, which is related to "G06Q 10/02". The lines of ② and ③ include production workflows and optimizing project, etc., which correspond to "G06Q 10/06" and "G06Q 10/04", respectively. The path labeled as ④ depicts some techniques of inventory and service management, which is relevant to "G06Q 10/08". These three evolution paths give us a general understanding of how technologies evolve with respect to the corresponding categories. These results have been reviewed and assessed by domain experts.

One interesting phenomenon in Figure 5 is the path of ⑤, which describes the technologies of health care management, such as medical intelligence, patient treatment, etc. From Table 3 we cannot find a mapping between this topic and the available codes. We further check the detailed assignments of classification codes to the patents along this line, and find that besides "G06Q 10/00", the patents are all assigned to the code "G06Q 50/00", which includes the classification of health care and patient record management. It somehow indicates that "G06Q 50/00" is more suitable to these patents rather than "G06Q 10/00". The analysts may be able to obtain more insights by using our proposed framework.

5.3.2 A Case Study on PatentTrace

PatentTrace formalizes the problem of tracing back a given technology/patent. The purpose is to trace a given patent document back to its ancestors to investigate what techniques that the given patent utilizes. To validate the proposed solution for this problem, we use the patent data under the international classification code of "G02F 1/1335", which represents the structural association of optical de-

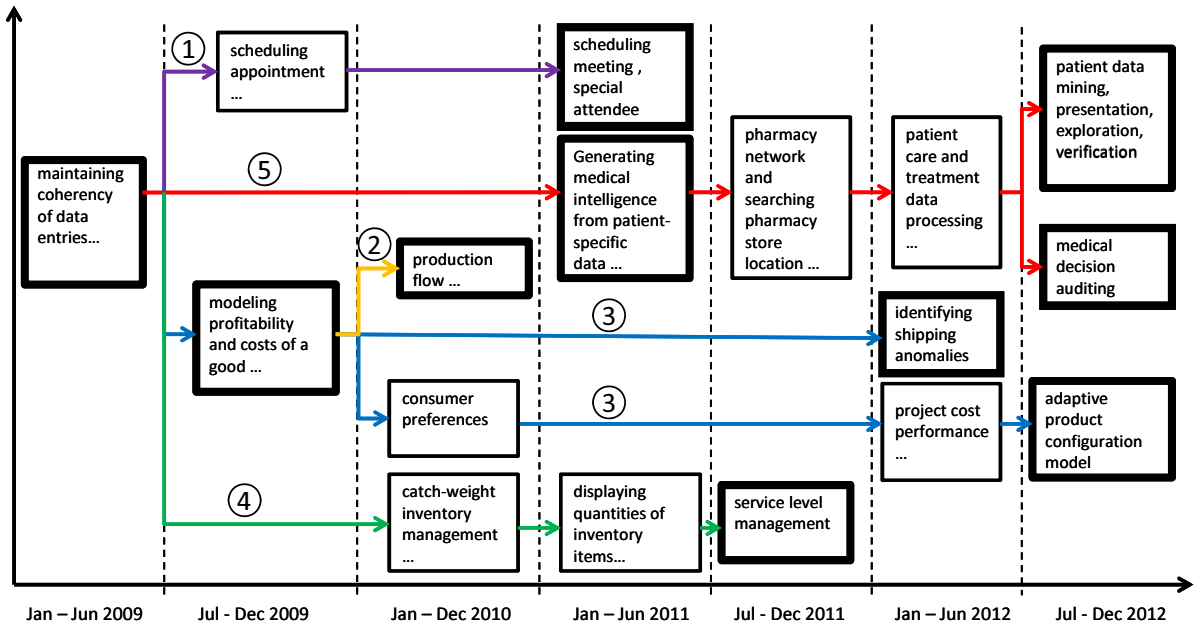

Figure 5: A case study of PatentLine.

vices, e.g., polarisers and reflectors. The data contains 3,080 patent documents. The query patent used in this case study is US8269915, which is related to a type of liquid crystal display apparatus (LCD), and was filed in 2008. Our goal is to examine what techniques are adopted in this product and how these techniques evolve to the product.

We treat US8269915 as a query and run the query-focused version of Algorithm 1 (limiting the number of dominators to be 20). We then run Algorithm 2 on the generated multi-view patent graph. The resulted back tracing Steiner tree is demonstrated in Figure 6. Similar to the case study of PatentLine, we only list the keywords of the title of patents for each patent document. The **bold** rectangles denote the dominators identified by query-focused MCDS. The X-axis represents the filing dates of patents.

This type of LCD contains two major components, i.e., the display and optical components. Our proposed solution to PatentTrace has successfully identified these two components (as depicted in Figure 6). For the display component, it involves polarized lighting plate (as indicated in the line of ②) and color filtering array (described by the line of ①). For the optical component, it consists of three major devices, i.e., optical film (③), prism sheet (④), and back-light unit (⑤). The figure outlines the major constituent parts of LCD, and describes how related techniques evolve to the corresponding components. For example, as indicated by line ③, the function of the optical film was originally fulfilled by birefringent retardation film, and then changed to reflective optical sheet, and finally laminated optical film. These results have been validated by patent analysts.

5.3.3 *A Case Study on* PatentLink

In practice, the linkage between two technologies is often achieved by technology evolution or technology application. The goal of PatentLink is to discover the details of evolution or application, in which the identified key patents serve to the ties that bind the technologies together. This would be very helpful for patent analysts to effectively understand the linkage between technologies.

To validate the efficacy of our solution to PatentLink, we present a case study on a collection of patent documents under the international classification code of "G03F 7/00", which represents the photomechanical production of textured or patterned surfaces. This data set contains 2,922 granted patents. We try to find the linkage between the patents US7771916 and US8053172. The former describes a polymerizable composition, which was filed in 2004; the latter proposes a method of forming a photoresist pattern using the photoresist composition, which was filed in 2008. The polymeriable composition is not directly used in the latter patent.

The experimental setup is similar to the one of PatentTrace. The resulted center-piece subgraph is depicted in Figure 4. There are 4 dominators falling in between the filing time period of the two query patents. With the help of patent analysts, we can identify several interesting paths that reflect the technology evolution/application. For example, the path of the dotted line indicates how the technique of polymerizable composition evolves to the one of photresist composition, connected by the technique of photolithography in ①.

6. CONCLUSION AND FUTURE WORK

In this paper, we study the problem of identifying dominating technologies using granted patent documents. Based on the analysis of domain characteristics of patents, we propose a unified framework, called PatentDom, to detect key patents from a large number of patents in a structural way. We formulate the problem as the minimum-cost dominating set problem, and employ graph-based optimization approaches to solve this problem. We further present potential applications of the proposed framework, including outlining the technology evolution of a particular domain (PatentLine), tracing a given technique to prior technolo-

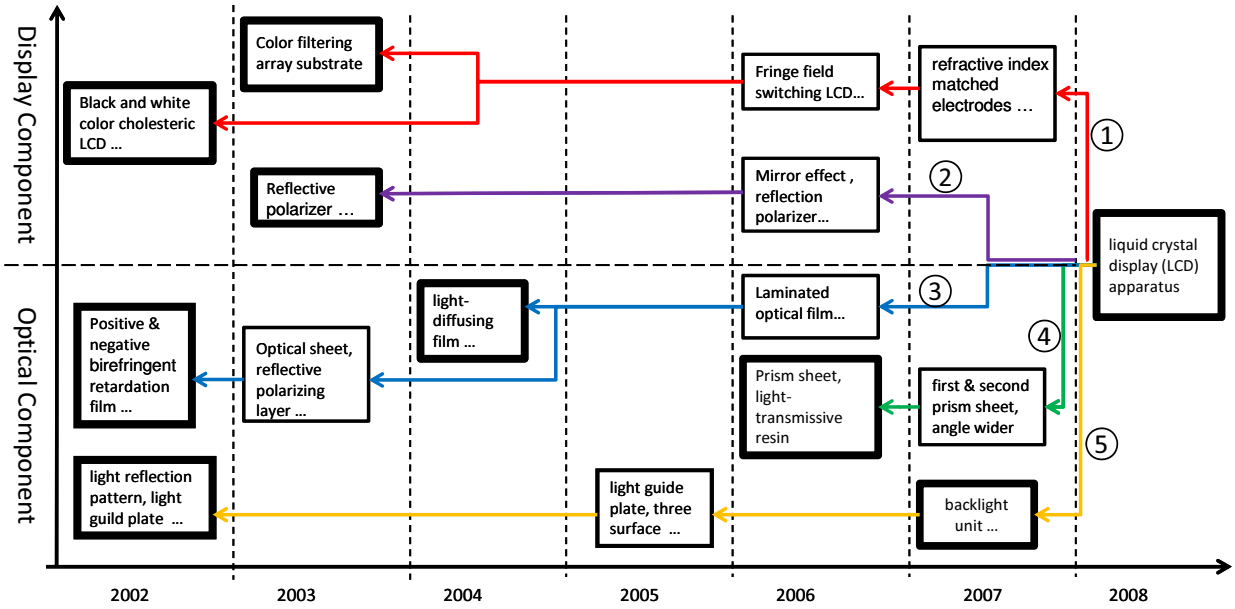

Figure 6: A case study of PatentTrace.

gies (PatentTrace), and mining the technical connection of two given patent documents (PatentLink). Simple yet effective graph-based approaches are proposed based on the identified key patents as well as the requirements of the corresponding applications. Extensive empirical evaluation and case studies on a collection of US patents demonstrate the efficacy and effectiveness of our proposed framework.

In our proposed framework, the cost of a vertex (patent) is defined based on the content and citation counts of the corresponding patent. It is interesting to extend it using external resources, such as patent examination results [10], patent maintenance decisions [12], and court judgments [17]. These resources explicitly indicate the relative importance of the patents, and hence are helpful to refine the definition of the cost. Further, to construct the multi-view patent graph, we utilize the content from claims to calculate the similarity. Due to the complex structure of patent documents as well as the diverse writing styles, the similarity may not represent the actual proximity between patents. We plan to explore semantic methods to improve the rationality of the edge weight in the undirected part of the graph.

The three applications introduced in Section 4 are all exploratory studies. In the domain of patent analysis, these applications are able to help patent analysts quickly identify the expected information without too much human effort, and make the corresponding decisions. It is worthy to provide quantitative measures to evaluate the generated results based on the requirement of the applications. In addition, we also plan to discover more applications/problems that can be solved using the dominating patents identified by PatentDom. Further, to ease the understanding, an interesting direction is to explore ways of visualizing the generated tree/graph based structures of patent documents.

ACKNOWLEDGMENT

The work was supported in part by the National Science Foundation under grants DBI-0850203, CNS-1126619, and IIS-1213026, the U.S. Department of Homeland Security under Award Number 2010-ST-06200039, the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001, and the Army Research Office under grants W911NF-10-1-0366 and W911NF-12-1-0431.

7. REFERENCES

[1] L. Bolelli, Ş. Ertekin, and C. L. Giles. Topic and trend detection in text collections using latent dirichlet allocation. In *Advances in Information Retrieval*, pages 776–780. 2009.

[2] L. Bolelli, S. Ertekin, D. Zhou, and C. L. Giles. Finding topic trends in digital libraries. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries*, pages 69–72. ACM, 2009.

[3] A. F. Breitzman and M. E. Mogee. The many applications of patent analysis. *Journal of Information Science*, 28(3):187–205, 2002.

[4] M. Charikar, C. Chekuri, T.-y. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li. Approximation algorithms for directed steiner problems. In *Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, pages 192–200. ACM, 1998.

[5] X. Cheng, X. Huang, D. Li, W. Wu, and D.-Z. Du. A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. *Networks*, 42(4):202–208, 2003.

[6] T. U. Daim, G. Rueda, H. Martin, and P. Gerdsri. Forecasting emerging technologies: Use of bibliometrics and patent analysis. *Technological Forecasting and Social Change*, 73(8):981–1012, 2006.

[7] A. Fujii. Enhancing patent retrieval by citation analysis. In *Proceedings of ACM SIGIR*, pages 793–794. ACM, 2007.

[8] M. A. Hasan, W. S. Spangler, T. Griffin, and A. Alba. Coa: finding novel patents through text analysis. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1175–1184. ACM, 2009.

[9] Q. He, B. Chen, J. Pei, B. Qiu, P. Mitra, and L. Giles. Detecting topic evolution in scientific literature: how can citations help? In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 957–966. ACM, 2009.

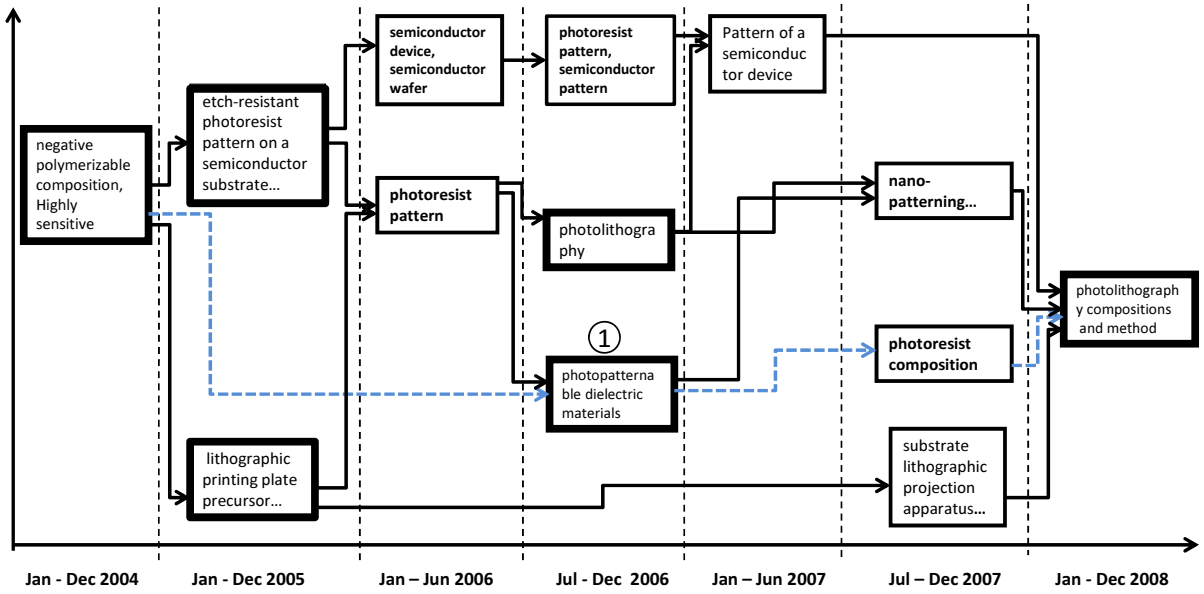

Figure 7: A case study of PatentLink.

[10] S. Hido, S. Suzuki, R. Nishiyama, T. Imamichi, R. Takahashi, T. Nasukawa, T. Idé, Y. Kanehira, R. Yohda, T. Ueno, et al. Modeling patent quality: A system for large-scale patentability analysis using text mining. *Journal of Information Processing*, 20(3):655–666, 2012.

[11] P. Hu, M. Huang, P. Xu, W. Li, A. K. Usadi, and X. Zhu. Finding nuggets in ip portfolios: core patent mining through textual temporal analysis. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1819–1823. ACM, 2012.

[12] X. Jin, S. Spangler, Y. Chen, K. Cai, R. Ma, L. Zhang, X. Wu, and J. Han. Patent maintenance recommendation with patent information network model. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 280–289. IEEE, 2011.

[13] V. Kann. *On the approximability of NP-complete optimization problems*. PhD thesis, Royal Institute of Technology Stockholm, 1992.

[14] R. M. Karp. *Reducibility among combinatorial problems*. 1972.

[15] L. Li, D. Wang, C. Shen, and T. Li. Ontology-enriched multi-document summarization in disaster management. In *Proceedings of ACM SIGIR*, pages 819–820. ACM, 2010.

[16] C. Lin, C. Lin, J. Li, D. Wang, Y. Chen, and T. Li. Generating event storylines from microblogs. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 175–184. ACM, 2012.

[17] Y. Liu, P. Hseuh, R. Lawrence, S. Meliksetian, C. Perlich, and A. Veen. Latent graphical models for quantifying and predicting patent quality. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1145–1153. ACM, 2011.

[18] J. Michel and B. Bettels. Patent citation analysis. a closer look at the basic input data from patent search reports. *Scientometrics*, 51(1):185–201, 2001.

[19] D. Newman, A. Asuncion, P. Smyth, and M. Welling. Distributed algorithms for topic models. *The Journal of Machine Learning Research*, 10:1801–1828, 2009.

[20] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability pcp characterization of np. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 475–484. ACM, 1997.

[21] B. Shaparenko, R. Caruana, J. Gehrke, and T. Joachims. Identifying temporal patterns and key players in document collections. In *Proceedings of the IEEE ICDM Workshop on Temporal Data Mining: Algorithms, Theory and Applications (TDM-05)*, pages 165–174, 2005.

[22] C. Shen and T. Li. Multi-document summarization via the minimum dominating set. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 984–992. Association for Computational Linguistics, 2010.

[23] H. Tong and C. Faloutsos. Center-piece subgraphs: problem definition and fast solutions. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 404–413. ACM, 2006.

[24] H. Tong, C. Faloutsos, and Y. Koren. Fast direction-aware proximity for graph mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 747–756. ACM, 2007.

[25] D. Wang, T. Li, and M. Ogihara. Generating pictorial storylines via minimum-weight connected dominating set approximation in multi-view graphs. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 683–689. AAAI, 2012.

[26] H.-C. Wu, H.-Y. Chen, and K.-Y. Lee. Unveiling the core technology structure for companies through patent information. *Technological forecasting and social change*, 77(7):1167–1178, 2010.

[27] S.-C. Wu and H.-Y. Chen. Recognizing the core technology capabilities for companies through patent co-citations. In *Industrial Engineering and Engineering Management, 2007 IEEE International Conference on*, pages 2081–2085. IEEE, 2007.

[28] L. Zhang, L. Li, T. Li, and Q. Zhang. Patentline: analyzing technology evolution on multi-view patent graphs. In *Proceedings of ACM SIGIR*, pages 1095–1098. ACM, 2014.

[29] L. Zhang and T. Li. Data mining applications in patent analysis. In *Data mining where theory meets practice*, pages 392–416. Xiamen University Press, 2013.

[30] D. Zhou, X. Ji, H. Zha, and C. L. Giles. Topic evolution and social interactions: how authors effect research. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 248–257. ACM, 2006.