Available online at www.sciencedirect.com

ScienceDirect

Information Processing and Management 43 (2007) 1216–1247

INFORMATION PROCESSING & MANAGEMENT

www.elsevier.com/locate/infoproman

Text mining techniques for patent analysis

Yuen-Hsien Tseng [a,*], Chi-Jen Lin [b], Yu-I Lin [c]

[a] *National Taiwan Normal University, No. 162, Sec., 1, Heping East Road, Taipei 106, Taiwan, ROC*
[b] *WebGenie Information Ltd., B2F., No. 207-1, Sec., 3, Beisin Road, Shindian City, Taipei 231, Taiwan, ROC*
[c] *Taipei Municipal University of Education, 1, Ai-Kuo West Road, Taipei 100, Taiwan, ROC*

Received 1 September 2005; accepted 29 May 2006
Available online 26 January 2007

Abstract

Patent documents contain important research results. However, they are lengthy and rich in technical terminology such that it takes a lot of human efforts for analyses. Automatic tools for assisting patent engineers or decision makers in patent analysis are in great demand. This paper describes a series of text mining techniques that conforms to the analytical process used by patent analysts. These techniques include text segmentation, summary extraction, feature selection, term association, cluster generation, topic identification, and information mapping. The issues of efficiency and effectiveness are considered in the design of these techniques. Some important features of the proposed methodology include a rigorous approach to verify the usefulness of segment extracts as the document surrogates, a corpus- and dictionary-free algorithm for keyphrase extraction, an efficient co-word analysis method that can be applied to large volume of patents, and an automatic procedure to create generic cluster titles for ease of result interpretation. Evaluation of these techniques was conducted. The results confirm that the machine-generated summaries do preserve more important content words than some other sections for classification. To demonstrate the feasibility, the proposed methodology was applied to a real-world patent set for domain analysis and mapping, which shows that our approach is more effective than existing classification systems. The attempt in this paper to automate the whole process not only helps create final patent maps for topic analyses, but also facilitates or improves other patent analysis tasks such as patent classification, organization, knowledge sharing, and prior art searches.

Keywords: Summarization; Phrase extraction; Co-word analysis; Clustering; Topic mapping

1. Introduction

Patent documents contain important research results that are valuable to the industry, business, law, and policy-making communities. If carefully analyzed, they can show technological details and relations, reveal business trends, inspire novel industrial solutions, or help make investment policy (Campbell, 1983; Jung, 2003). In recent years, patent analysis had been recognized as an important task at the government level in

[*] Corresponding author. Tel.: +886 2 23215131; fax: +886 2 23222009.
E-mail addresses: samtseng@ntnu.edu.tw (Y.-H. Tseng), dan@webgenie.com.tw (C.-J. Lin), jg141@mail.jges.tpc.edu.tw (Y.-I. Lin).

0306-4573/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2006.11.011

Table 1
A typical patent analysis scenario

1. **Task identification**: define the scope, concepts, and purposes for the analysis task
2. **Searching**: iteratively search, filter, and download related patents
3. **Segmentation**: segment, clean, and normalize structured and unstructured parts
4. **Abstracting**: analyze the patent content to summarize their claims, topics, functions, or technologies
5. **Clustering**: group or classify analyzed patents based on some extracted attributes
6. **Visualization**: create technology-effect matrices or topic maps
7. **Interpretation**: predict technology or business trends and relations

some Asian countries. Public institutions in China, Japan, Korea, Singapore, and Taiwan have invested various resources in the training and performing of the task of creating visualized results for ease of various analyses (Liu, 2003). For example, the Korean Intellectual Property Office plans to create 120 patent maps for different technology domains in the next 5 years (Bay, 2003).

Patent analysis or mapping requires considerable effort and expertise. For example, Table 1 shows a typical patent analysis scenario which is based on the training materials designed for patent analysts, such as those in Chen (1999). As can been seen, these processes require the analysts to have a certain degree of expertise in information retrieval, domain-specific technologies, and business intelligence. This multi-discipline requirement makes such analysts hard to find or costly to train. In addition, patent documents are often lengthy and rich in technical and legal terminology. To read and analyze them may consume a lot of time even for experts. Automated technologies for assisting analysts in patent processing and analysis are thus in great demand.

A patent document contains dozens of items for analysis; some are structured, meaning they are uniform in semantics and in format across patents such as patent number, filing date, or assignees; some are unstructured, meaning they are free texts of various lengths and contents, such as claims, abstracts, or descriptions of the invention. The visualized results from patent analysis are called *patent graphs* if they are from the structured data and *patent maps* if they are from the unstructured texts, although, loosely speaking, patent maps can refer to both cases.

Before their publication, patent documents are given one or more classification codes based on their textual contents for topic-based analysis and retrieval. However, these pre-defined categories may be either too broad or not meet the goal for a particular analysis. Self-developed classification systems are often needed. For example, Table 2 shows part of a patent map created by analyzing 92 patent documents (Mai, Hwang, Chien, Wang, & Chen, 2002). These patents, issued before February 19, 2002, are the search results of the keyword: "carbon nanotube" from the database of USPTO (United States Patent & Trademark Office). In Table 2, manually assigned categories regarding the technology aspects of the patents are listed in rows and those

Table 2
Part of the technology-effect matrix for "Carbon Nanotube" from 92 US patents

Technology		Effect (function)			
		Material	Performance		Product
		Carbon nanotube	Purity	Electricity	FED
Manufacture	Gas reaction	5346683 6129901 ...	6181055 6190634	6221489	6232706
	Catalyst	5424054 5780101 ...	6333016		6339281
	Arc discharging	5424054	6190634 6331262	5916642	5916642
Application	Display			6346775	5889372 5967873 ...

regarding the effect (or function) aspects of the patents are listed in columns. The patent IDs are then assigned to the cells of the *technology-effect matrix* based on their contents. With this map, patent relations and distributions are revealed among these aspects. This information can be used to make decisions about future technology development (such as seeking chances in those sparse cells), inspire novel solutions (such as by understanding how patents are related so as to learn how novel solutions were invented in the past and can be invented in the future), or predict business trends (such as by showing the trend distribution of major competitors in this map).

Depending on the goals, the created map may need to reflect the relations among some machine-identified topics. In such a case, only the relative distance between each topic in the map is relevant, while the absolute position and orientation of each topic does not matter. As an example, Fig. 1 shows a *topic map* analyzed based on the above 92 carbon nanotube patents, where each circle denotes an identified topic, the size of the circle denotes the number of patents belonging to the topic, and the number in the circle corresponds to the topic ID. Some of the topic titles and their IDs are shown in Table 3.

Creating and updating such maps requires a lot of human effort. As in the above ‘‘carbon nanotube’’ map (called *CNT map*), five specialists spent more than one month in analyzing about one hundred patents. Although it may go unnoticed, such efforts indeed involve some text mining processes such as text segmentation, summary extraction, keyword identification, topic detection, taxonomy generation, term clustering, and document categorization. As shown in Table 1, this patent analysis scenario is quite similar to the general text

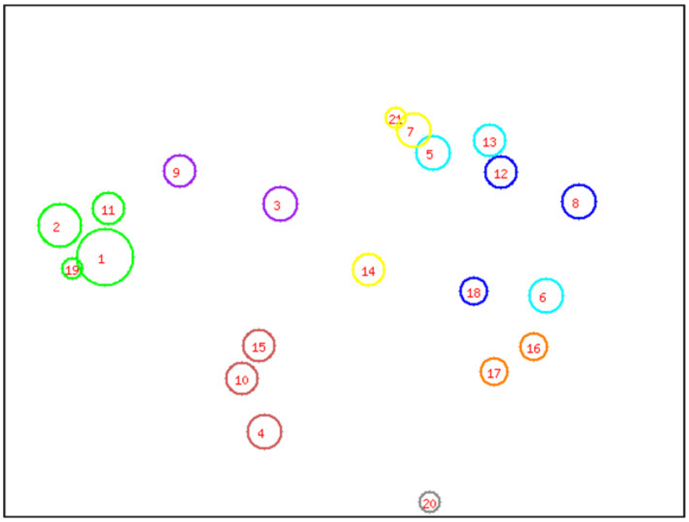


Fig. 1. Topic map from 92 patents about carbon nanotube.

Table 3
Part of the topics and their titles organized hierarchically for the map in Fig. 1

25 docs.: 0.228054 (emission: 180.1, field: 177.2, emitter: 157.1, cathode: 108.4, field emission: 88.0)
 + 23 docs.: 0.424787 (emitter: 187.0, emission: 141.9, field: 141.4, cathode: 129.0, field emission: 104.7)
  + 19 docs. : 0.693770 (emitter: 139.7, field emission: 132.0, cathode: 96.0, electron: 67.1, display: 61.9)
   + ID = **2: 7 docs.,0.09 (cathode: 0.58, source: 0.56, display: 0.50, field emission: 0.45, vacuum: 0.43)**
   + ID = **1: 12 docs.,0.07 (emitter: 0.67, emission: 0.60, field: 0.57, display: 0.40, cathode: 0.38)**
  + ID = **11: 4 docs.,0.13 (chemic vapor deposition: 0.86, sic: 0.56, grow: 0.44, plate: 0.42, thickness: 0.42)**
 + ID = **19: 2 docs.,0.21 (electron-emissive: 1.00, carbon film: 0.70, compromise: 0.70, emissive material** ...)

13 docs.: 0.240830 (energy: 46.8, circuit: 34.0, junction: 33.3, device: 26.0, element: 24.9)
 + 9 docs.: 0.329811 (antenna: 31.0, energy: 29.5, system: 29.4, electromagnetic: 25.0, granular: 20.6)
  + ID = **4: 5 docs.,0.07 (wave: 0.77, induc: 0.58, pattern: 0.45, nanoscale: 0.44, molecule: 0.35)**
  + ID = **15: 4 docs.,0.12 (linear: 0.86, antenna: 0.86, frequency: 0.74, optic antenna: 0.70,** ...**)**
 + ID = **10: 4 docs.,0.06 (cool: 0.70, sub-ambient: 0.70, thermoelectric cool apparatus: 0.70, nucleate: 0.70,** ...)

mining process commonly discussed in the literature, such as those in Hearst (1999), Losiewicz, Oard, and Kostoff (2000).

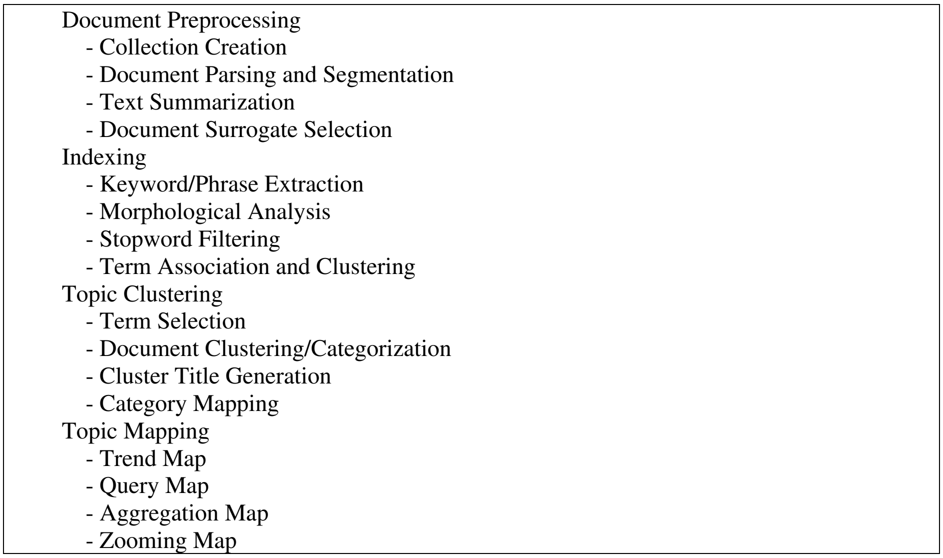
Text mining, like data mining or knowledge discovery (Fayyad, Piatetsky-Shapiro, Smyth, & Uthurasamy, 1996), is often regarded as a process to find implicit, previously unknown, and potentially useful patterns from a large text repository. In practice, the text mining process involves a series of user interactions with the text mining tools to explore the repository to find such patterns. After supplemented with additional information and interpreted by experienced experts, these patterns can become important intelligence for decision-making.

The purpose of this paper is to present a text-mining approach that help automate the patent analysis scenario discussed above, based on the patent documents from USPTO. In particular, we propose and implement a text mining method for each technical step in Table 1. The adopted methodology is first introduced in Section 2. The technical details of each method are described in Section 3. Evaluation of some critical techniques is conducted in Section 4. In Section 5, an example of analyzing a set of patents is given. Section 6 discusses the implications and related work. Finally Section 7 concludes this paper.

2. A general methodology

Patent analyses based on structured information such as filing dates, assignees, or citations have been the major approaches in practice and in the literature for years (Archibugi & Pianta, 1996; Be'de'carrax & Huot, 1994; Ernst, 1997; Lai & Wu, 2005). These structured data can be analyzed by bibliometric methods, data mining techniques, or well-established database management tools such as OLAP (On-Line Analytical Processing) modules. Recently, there has been an interest in applying text mining techniques to assist the task of patent analysis and patent mapping (ACL-2003 Workshop on Patent Corpus Processing, 2003; Fattori, Pedrazzi, & Turra, 2000 Lent, Agrawal, & Srikant, 2000; Fattori et al., 2003; Lent et al., 1997; Yoon & Park, 2004). A well-utilization of the full texts in the patent documents may complement the interpretations derived from the bibliometric analysis.

Therefore, based on the patent analysis scenario introduced above, a text mining methodology specialized for full-text patent analysis is proposed and shown in Fig. 2. First, full patent documents relevant to the analysis purpose are collected. This may involve a repeated process of devising a set of query terms (query formulation), searching a couple of patent databases (collection selection), filtering undesired patents (relevance judgment), and downloading patents for local analysis (data crawling). Depending on the analysis purpose, the step can be as easy as, for example, fetching all the patents under some IPC (International Patent



Document Preprocessing
- Collection Creation
- Document Parsing and Segmentation
- Text Summarization
- Document Surrogate Selection

Indexing
- Keyword/Phrase Extraction
- Morphological Analysis
- Stopword Filtering
- Term Association and Clustering

Topic Clustering
- Term Selection
- Document Clustering/Categorization
- Cluster Title Generation
- Category Mapping

Topic Mapping
- Trend Map
- Query Map
- Aggregation Map
- Zooming Map

Fig. 2. The general text mining methodology for patent analysis.

Classification) categories and within some year limits, or as hard as searching patent documents relevant to a news story to understand current status of technologies mentioned in the story or searching all the "prior arts" that can invalidate competitors' patents. The later, called *technology survey* and *invalidity search*, respectively, are two of the main tracks in patent retrieval evaluation of NTCIR Workshop 3 (Iwayama, Fujii, Kando, & Marukawa, 2006) and 4 (Fujii, Iwayama, & Kando, 2004). Since this issue deserves a monograph, it is beyond the scope of our discussion. Our method assumes that a set of patents has been carefully prepared.

Next, each document in the collection is parsed and segmented. Structured data parsed from the patent documents are saved into a DBMS system for ease of management and unstructured data (the full texts) are segmented into smaller units for summarization. Take the patent documents from the USPTO as examples, the structured data include: filing date, application date, assignees, UPC (US Patent Classification) codes, IPC codes, and others, while the unstructured segments include: title, abstract, claims, and description of the invention. The description of the invention can be further segmented into field of the invention, background, summary, and detailed description, although some patents may not have all these segments.

In our analysis process, the title and abstract are not the only textual parts used, as is often the case in analyzing scientific publications. Thanks to the consistent format of patent documents, all the summaries of the above-mentioned segments are used. Summarization of the segments (document surrogate) can yield concise representation. It may not only facilitate the sharing and re-use of the analyzed patents among analysts, but it also speedups later automated processing due to less textual data and possibly yields higher effectiveness due to the elimination of less-focused snippets. Although perfect machine-derived summarization is hard to define and achieve, simple methods based on sentence ranking and selection often yield sufficient performance for some text mining tasks. For example, in a patent classification experiment using Naïve Bayes, KNN, and SVM as classifiers, Fall, Torcsvari, Benzineb, and Karetka (2003) shows that even using only the first 300 words from the abstract, claims, and description sections, the performance is better than those using the full texts regardless of which classifiers are used.

From the set of document surrogates, keywords and phrases are extracted. The resultant terms are further filtered by a stopword list and by some frequency criteria. The goal is to extract high-quality terms for indexing and analysis. Yet ideal indexing includes term stemming (morphological analysis) and term clustering so that the terms in various forms corresponding to the similar concepts are associated together in the same set. This not only reduces the size of the vocabulary for efficient analysis, but also decreases the vocabulary mismatch problem for effective clustering.

With these concise representations in terms and documents, various clustering algorithms can be applied to identify the concepts underlying the collection. These concepts can be further clustered into topics, which in turn can be clustered into categories or domains. Here the distinction among concepts, topics, and categories is not important. As long as human analysts can recognize the knowledge structures underlying the collection, the multi-stage clustering process can stop at any level of granularity.

To show the detected knowledge structures, various forms of representation can be used, such as the folder tree, the 2-D matrix, or the topic map mentioned above. From the folder tree, the results from each of the multi-stage clustering can be directly shown for exploring. From the topic map, several types of maps for visual analysis can be derived by combining other structured data such as time, assignees, number of patents, etc. A general visualization idea is that: for data visualization, occurrence frequencies of dependent variables (such as number of patents or assignees) are plotted against independent variables whose values are arrayed along a (regularly) varying base (such as patent application years). As such, for text-based visualization, the values of the independent variable become the identified clusters based on text similarities, with the dependent variables being any structured data. All the values of these variables can be filtered, pivoted, or sliced for various analysis purposes.

Examples of the maps derived from the above idea are: (1) Trend map: it can be further divided into 2 subtypes – growth map which shows how topics grow in size (number of patents) with time and evolution map which shows how topics evolve (change in size and relation) over time. (2) Query map: showing only those patents satisfying some query conditions in each topic. The sizes of and the similarities among these topics can be re-calculated based on the filtered patents in each cluster to reveal new relationships or patterns among the topics. (3) Aggregation map: showing those aggregated results based on some specified attributes. Examples are the shares of the top-three assignees distributed in each topic. (4) Zooming map: showing the details or overview of the selected part in the map.

3. Technique details

This section presents the details of the proposed techniques in dealing with patent documents. Each technical step in the above methodology is addressed in sequence. The rationales behind these techniques are discussed. Efficiency, effectiveness, robustness, and degree of automation are taken into consideration. For better presenting the proposed techniques, a number of examples are given when necessary.

3.1. Text segmentation

The textual content of a patent document from USPTO is in HTML format and contains title, abstract, claims, and description. The description, the main body of the detailed content, often have sub-sections with titles in uppercase, such as FIELD OF THE INVENTION, BACKGROUND, SUMMARY OF THE INVENTION, and DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT. Although some patents may have more or fewer such sub-sections or have slightly different title names, most patents do follow this style. Thus a regular expression matcher is devised to extract each of these segments. The method takes advantage of the rule that each sub-section's title is in a single line paragraph separated by two HTML tags: "〈BR〉〈BR〉". After splitting the paragraphs based on these tags, a set of Perl expressions: (/Abstract/i, /Claims/i, /FIELD/i, /BACKGROUND|Art/i, /SUMMARY/i, /DESCRIPTION|EMBODIMENT/i) are used to match the patent segments. An exception is that if FIELD OF THE INVENTION is not detected, the first text paragraph of the BACKGROUND segment is extracted as the FIELD segment. As long as non-relevant single-line paragraphs are properly filtered (those that are too long, too short, and do not contain the above title words), the ratio of false drops (the cases of incorrect detection of the segment titles) can be kept to a minimum. The whole process is quite ad hoc. The details can be varied slightly without affecting the effectiveness very much. We have observed that our colleagues achieved performance similar to ours in this task based on our guidelines, rather than on our source code.

3.2. Text summarization

Automatic summarization techniques have been widely explored in recent years (Document Understanding Conferences). They can be mainly divided into two approaches: abstraction and extraction. In abstraction, natural language understanding techniques are applied to analyze sentential semantics and then to generate concise sentences with equivalent semantics. Sophisticated techniques such as sense disambiguity or anaphoric resolution and large human-maintained resources may be applied. In extraction, statistical techniques are applied to rank and select the text snippets as a summary. In our method, the extraction approach is adopted for its relatively low cost and high robustness across technical domains. It is divided into three processing stages: sentence breaking, sentence weighting, and sentence selection and presentation.

Our extraction-based method takes sentences as the smallest units for summarization. As such, each segment is broken up by simply judging a period and question mark as a sentence break. But care has to be taken for exceptions involving floating-point digits, mathematical or chemical expressions (such as "X.sub.i"), various abbreviations (such as " Fig. 1"), or sentences within sentences (such as those containing citations).

Next, each sentence is weighted by the number of keywords, title words, and clue words it contains. Furthermore, the position of the paragraph containing the sentence in a segment and the position of the sentence in the containing paragraph are considered as more important information for weighting. (Recall that paragraphs in the USPTO patent documents are separated by the "〈BR〉〈BR〉" HTML tags.) Here the keywords are those maximally repeated patterns extracted by the algorithm to be described. The title words are those non-stopwords that occur in the title of a patent document. As to the clue words, they are a list of about 25 special words that reveal the intent, functions, purposes, or improvements of the patent. These words are prepared by several patent analysts based on their experiences and are listed in Appendix A for reference. In our implementation, the weight of a sentence is calculated as follows:

$$weight(S) = \left(\sum_{w \in keywords\_or\_titlewords} tf_w + \sum_{w \in cluewords} avgtf \right) \times \mathrm{FS} \times P$$

where tf_w is the term frequency (occurrence frequency in a segment) of word w which occurs in sentence S, $avgtf$ is the term frequency averaged over all keywords and title words in the segment, and FS and P are the position weights. Heuristically, FS is set to 1.5 if S is the first sentence of any paragraph, and 1 otherwise, while P is set to 2 if sentence S is in the first two paragraphs in a segment, 4 if it is in the last two paragraphs and 1 otherwise. These values for FS and P are specialized for the background segment. Other segments can be further tuned empirically or based on some learning method if training data is available (Kupiec, Pedersen, & Chen, 1995).

Sentences are then sorted by their weights in decreasing order. By giving a desired summary length or ratio, a number of top-ranked sentences are selected. They are combined together in their original order to result in a summary or they are highlighted in the original segment to provide better readability.

Appendix B shows the background segment of a patent with the best sentences underlined. Two sets of summary are highlighted: one from the Microsoft Word's Auto Summarization, which is in italic font; the other from the above method, which is in boldface. Both methods were confined to select the best 3–4 sentences. As can be seen, Word's summary is more general in introducing the background or current status, while ours directly specifies the motivation of the patent and the problem to be solved.

3.3. Stopwords and stemming

Term frequency (TF) and inverse document frequency (IDF) are the two parameters used in filtering terms. Low TF and DF terms are often removed from the indexing of a collection. However, using them alone does not prevent undesired terms such as function words from being calculated. A list of over 250 stop words from (van Rijsbergen) is used in our text processing. After analyzing a set of patent documents, another 200 words chosen by hand were added to the list. They are adverbs mostly. A few are verbs, nouns, and adjectives commonly seen in a patent document.

To better match concepts among terms, words are stemmed based on Porter's algorithm (Porter, 1980). However, the algorithm is so aggressive in removing the word's suffix such that stemmed words become hard to read for analysts. We therefore modify the algorithm to remove only simple plurals and general suffixes (such as regular paste tense).

3.4. Keyword and phrase extraction

In patent analysis, single words alone are often too general in meanings or ambiguous to represent a concept. Multi-word phrases can be more specific and desirable. Thus besides important single words, multi-word phrases are also extracted for later use. However, keywords or key-phrases have no lexical boundaries in texts, making them a challenge to identify. Other studies had used statistical approaches based on corpus-wide statistics (Choueka, 1988), while ours applies a simple and fast key term extraction algorithm document by document. The algorithm works with the help of a stopword list alone. Other resources, such as corpora, lexicons, or dictionaries are not required. As such, it is readily applicable to any knowledge domains without much parameter tuning or resource acquisition.

In this algorithm, the text to be processed is first split into a series of words. The algorithm then repeatedly merges back nearby words based on three simple merging, dropping, and accepting rules. Maximally repeated strings in the text are thus extracted as keyword candidates. By maximally, we mean that either the repeated strings are the longest ones or they occur more often than the longer strings that contain them. For example, a repeated term "public high school" in a certain document may be extracted without extracting "public high" or "high school", as they are exact substrings of the longer term. Only when "high school" occurs more often than "public high school" (in such a case we may say: "high school" subsumes "public high school"), can "high school" be possibly extracted. The resultant candidates are then subject to a filtering process. A precision-oriented rule may remove candidates containing any stopwords. A recall-oriented rule may only the

```
1. Convert the input text into a LIST of words.
2. Do Loop
  2.1     Set MergeList to empty.
  2.2     Put a separator to the end of LIST as a sentinel and set the occurring frequency of the separator to 0.
  2.3     For I from 1 to NumOf(LIST) - 1 step 1, do
          2.3.1   If LIST[ I ] is the separator, Go to Label 2.3.
          2.3.2   If Freq(LIST[ I ]) > threshold and Freq(LIST[ I+1]) > threshold, then
                      Merge LIST[ I ] and LIST[ I +1] into Z.
                      Put Z to the end of MergeList.
                  Else
                      If Freq(LIST[ I ]) > threshold and LIST[ I ] did not merge with LIST[ I - 1], then
                          Save LIST[ I ] in FinalList.
                      If the last element of MergeList is not the separator, then
                          Put the separator to the end of MergeList.
          End of For loop
  2.4 Set LIST to MergeList.
  Until NumOf(LIST) < 2.
3. Filter terms in FinalList based on some criteria.
```

Fig. 3. The keyword extraction algorithm.

```
Example: Given an input string: BACDXAYCDBACD.
Let threshold=1, separator=x.
Step 1: Create a list of single tokens:
  LIST = (B:2, A:3, C:3, D:3, X:1, A:3, Y:1, C:3, D:3, B:2, A:3, C:3, D:3, x)
Step 2:
 After 1st iteration :
   MergeList = (BA:2, AC:2, CD:3, x, CD:3, DB:1, BA:2, AC:2, CD:3, x)
   FinalList = (A:3)
 After 2nd iteration :
   MergeList = (BAC:2, ACD:2, x, BAC:2, ACD:2, x)
   FinalList = (A:3, CD:3)
 After 3rd iteration :
   MergeList = (BACD:2, x, BACD:2, x)
   FinalList = (A:3, CD:3)
 After 4th iteration :
   MergeList = (x)
   FinalList = (A:3, CD:3, BACD:2)
```

Fig. 4. A running example of the algorithm, where the number following a semicolon denotes the occurring frequency of the associated string.

stopwords from the head and tail of the candidates recursively. Fig. 3 shows the algorithm. The rule for merging, dropping, and accepting terms is implicit expressed in the algorithm. Fig. 4 shows a running example, in which each capital letter denotes a word.

The above algorithm is based on the assumption that a document concentrating on a topic is likely to mention a set of strings a number of times. Many natural language documents have this property, including Chinese, Japanese, or even melody strings in music (Tseng, 1999). We found that a longest repeated string often is a correct word (or phrase), since its repetition provides evidence for decision on its left and right boundaries. Similarly, a repeated string that subsumes the others may also be a legal term. The sources of errors mainly come from the inadequate coverage of the stopword list.

3.5. Term association

There are a number of approaches to extract terms relevant to the same topics from the entire document collection. One commonly used heuristic rule is based on term co-occurrence (Salton, 1989). Given a collection of n documents, an inverted term-document structure is first constructed, where each term is denoted in a vector form whose elements are weights of the term in the documents, such as: $Tj = (d_{1j}, d_{2j}, \ldots, d_{nj})$. Similarities are then computed among all useful term pairs. A typical similarity measure is given by cosine function:

$$sim(T_j, T_k) = \sum_{i=1}^{n} d_{ij} d_{ik} \Bigg/ \sqrt{\sum_{i=1}^{n} d_{ij}^2 \sum_{i=1}^{n} d_{ik}^2}$$

If the weights of the terms were either 1 or 0, denoting the presence or absence of the terms in documents, the similarity becomes a value exactly proportional to the number of documents in which these two terms co-occur. With these pair-wise similarities, terms are clustered with some automatic processes.

However, the above method requires a lot of computations. With m distinct terms in a collection of n documents, this can be an $O(m^2n)$ algorithm (n steps to calculate similarity between any of $O(m^2)$ term pairs). Besides, terms co-occur in the same document may virtually have no relationship if they are far apart from each other in the text. Calculating their term similarities in this way may turn out to be a waste of computation power.

Therefore, we proposed another method that is far more efficient. The major difference of our method from the above is to limit the terms to be associated to those that co-occur in the same logical segments of a smaller text size, such as a sentence or a paragraph. Association weights are computed in this way for each document and then accumulated over all documents. This changes it into a roughly $O(nk^2s)$ algorithm, where k is the average number of selected keywords for association per document and s is the average number of sentences in a document.

Specifically, keywords or key terms extracted from each documents are first sorted in decreasing order of their term frequencies (TF), or TF × Term_Length, or other criterion such as TF × IDF (Inverse Document Frequency) if the entire collection statistics are known in advance. Then the first k terms are selected for association analysis. A modified Dice coefficient was chosen to measure association weights as:

$$wgt(T_{ij}, T_{ik}) = \frac{2 \times S(T_{ij} \cap T_{ik})}{S(T_{ij}) + S(T_{ik})} \times \ln(1.72 + S_i)$$

where S_i denotes the number of sentences (or paragraphs) in document i and $S(T_{ij})$ denotes in document i the number of sentences (paragraphs) in which term T_j occurs. Thus the first term is simply the Dice coefficient similarity. The second term $\ln(1.72 + S_i)$, where ln is the natural logarithm, is used to compensate for the weights of those terms in longer documents so that weights in documents of different length have similar range of values. This is because longer documents tend to yield weaker Dice coefficients than those generated from the shorter ones. Therefore, in a collection where relatively long documents may occur, the long ones should better be segmented into several shorter ones to prevent the term $\ln(1.72 + S_i)$ from being excessively large. Association weights larger than a threshold (1.0 in our implementation) are then accumulated over all the documents in the following manner:

$$sim(T_j, T_k) = \frac{\log(w_k \times n/df_k)}{\log(n)} \times \sum_{i=1}^{n} wgt(T_{ij}, T_{ik})$$

where df_k is the document frequency of term k and w_k is the width of k (i.e., number of constituent words). To build a term relation structure for later use, terms associated with the same one are sorted in decreasing order of their similarities. In our implementation, about 10–30 terms per document were selected for analysis and at most 64 co-occurred terms for each keyword were kept in this term relation structure.

Computation of the similarities among all term pairs can be carried out as the inverted index file for the entire collection is constructed. Weights of term pairs from each document are calculated and accumulated just like the index terms accumulating their document frequencies and postings (Frakes & Baeza-Yates, 1992). In this way, a global term relation structure can be obtained efficiently. As an example, for the 381,375 documents in the NTCIR-4 Chinese collection (469 MB of texts), it takes only 133 min on a notebook computer with a 1.7 GHz CPU, 512 Mega RAM, and 4500 RPM hard disk for indexing, keyword extraction, and term association computation (Tseng, Juang, & Chen, 2004). Compared to traditional approaches, this method improves the efficiency drastically while maintaining sufficient effectiveness.

However, in the light of ideal indexing, most of the term pairs obtained do not exhibit synonymous or near-synonymous relationship. They are only related to the same specific topics or events mentioned in the collection. To obtain closer relations among terms, they are further clustered based on their associated words. In our implementation, additional similarities among terms are calculated based on how many common associated terms are shared. The complete-link clustering algorithm is then used with a strict threshold to yield small clusters having high intra-similarities. As a result, terms which do not co-occur in any documents may be clustered together. It is noted that although Latent Semantic Indexing (LSI) (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990) exhibits this similar advantage, its time complexity is higher than the method presented here.

3.6. Topic clustering

Clustering is a powerful technique to detect topics and their relations in a collection. Various clustering methods have been proposed for years, such as HAC (Hierarchical Agglomerative Clustering) (Jain, Murthy, & Flynn, 1999), MDS (Multi-dimensional Scaling) (Kruskal, 1977), and SOM (Self-organization Map) (Kohonen, 1997). Many of their computer implementations either in standalone programs or in integrated packages can be found and downloaded from the Web for free (such as those in Karypis; Frank, Hall, &Trigg,Trig). Use of these tools requires a specification of how similarities between items are computed.

3.6.1. Document clustering

Specifically, in document clustering, each document is processed into a vector form, such as $d_i = (t_{i1}, t_{i2}, \ldots, t_{im})$, where t_{ij} is the weight of an indexed term j in document i. The effectiveness of clustering relies on (1) how terms are selected; (2) how they are weighted; (3) and how similarities are measured. From the experience of topic-based retrieval, term selection affects the performance most among these three factors; the other two are second (Tseng et al., 2004). In our implementation, term selection is based on the extracted key terms, associated terms, and TF or IDF filtering. Terms whose TF in a document is lower than a threshold are removed from that document. Terms whose DF is lower than a threshold or higher than another threshold are also removed. After this basic filtering for outliers, depending on how much information is to be used in the clustering, all the indexed terms can be retained; or only those extracted key terms are kept; or most strictly only those terms in the term relation structure are selected. Although intuitively the last choice may result in the highest-quality (and smallest) set of terms, topic-based retrieval experiments showed that long queries (having long description of information need including seemingly noisy terms) often lead to better performance than short queries (Fang, Tao, & Zhai, 2004). In text categorization tasks, excessive term selection and reduction may also hurt effectiveness in some real-world test collections (Bekkerman, El-Yaniv, Winter, & Tishby, 2001; Joachims, 1998; Yang & Pedersen, 1997). Therefore, selection of terms for clustering becomes, more or less, a try-and-error process. This was also observed in text visualization studies, such as (Booker et al., 1999) where parameters are provided for term selection. Although there are performance indices such as the overall intra-cluster and inter-cluster similarities to objectively compare different clustering results, these indices do not directly correspond to the quality of the clusters perceived by humans. From the analyst's point of view, the interpretability of the resulting clusters seems to be more associated with its quality. Therefore, to reduce un-expected efforts in tuning the clustering results, manual verification of selected terms is always recommended whenever it is possible.

As to the weighting of the selected terms, TF × IDF is used, which is combined with the cosine function to yield a similarity measure. This choice has a simple reason: the measure ranges from 0 to 1, a range compatible with those required by the mapping tools to be used. Other similarity measures, such as pivoted normalization (Singhal, Buckley, & Mitra, 1996) or OKAPI BM25 (a probability model) (Robertson & Walker, 1994), that perform better in other information retrieval tasks simply do not fit since their similarities are beyond this range and do not reflect the Euclidean distances among clusters.

3.6.2. Term clustering followed by document categorization

Another way for topic clustering is to perform term clustering followed by document categorization. Previous work used the term-document index for term clustering (Noyons & van Raan, 1998b). In our implementation, terms are clustered based on their co-occurred terms described above. Using all the terms (clustered and their co-occurred) as training data, documents can then be classified into these clusters based on some classification method, such as KNN (*K*-Nearest Neighbor). This approach has the advantage that it allows large volume of documents to be efficiently clustered.

3.6.3. Multi-stage clustering

In either document clustering or term clustering, multi-stage clustering can be applied to gradually identify the knowledge structures from concepts to topics and from topics to categories, or vice versa. Agglomeratively, the clusters which result from a previous stage are considered as super-documents in the current stage and clustering takes place as the same in the previous stage. Or, divisively, the collection is recursively partitioned into smaller sub-domains based on sampled documents and selected features. Then fine-grained clustering takes place in each sub-domain. Single-step clustering often leads to skewed document distributions among clusters. Therefore multi-stage clustering is applied in our analysis.

3.6.4. Cluster title generation

One important step to help analysts interpret the clustering results is to generate a summary title for each cluster. Again, this could be a summarization task. We adopt the extraction-based statistical approaches because of their robustness. One commonly used method is to select the most frequent terms in the cluster (Noyons & van Raan, 1998a; Yang, Ault, Pierce, & Lattimer, 2000). That is, terms are sorted by their total frequency in cluster (TFC) in decreasing order. Then the top *k* terms are selected as the cluster title. Here the TFC of a term in a cluster is defined as the sum of this term's TF in those documents belonging to the cluster.

While this method is simple, it runs the risk of selecting those frequent terms across clusters, making the clusters indistinguishable based on their titles. A remedy to this problem is the introduction of the correlation coefficient method, which computes the relatedness of term *T* with respect to category or cluster *C* according to the formula (Ng, Goh, & Low, 1997):

$$Co(T,C) = \frac{(\mathrm{TP} \times \mathrm{TN} - \mathrm{FN} \times \mathrm{FP})}{\sqrt{(\mathrm{TP}+\mathrm{FN})(\mathrm{FP}+\mathrm{TN})(\mathrm{TP}+\mathrm{FP})(\mathrm{FN}+\mathrm{TN})}}$$

where TP (True Positive), FP (False Positive), FN (False Negative), and TN (True Negative) denote the number of documents that belong or not belong to *C* while containing or not containing *T*, respectively, as shown in Table 4.

The correlation method is effective for large number of clusters having short documents in them. But it tends to select specific terms that are not generic enough for clusters having a few long documents, because it does not take TF into account. Therefore, we choose only those terms whose document frequency in a cluster exceeds half of the number of documents in that cluster for the correlation method in our implementation. We denote this method as $\mathrm{CC}_{0.5}$. Another remedy is to multiply the correlation coefficient with the TFC (i.e., CC × TFC) to rank the terms for title generation.

3.6.5. Mapping cluster titles to categories

The cluster titles generated by the above approach may not be topic-indicative enough to well summarize the contents of the clusters. One might need to map the identified clusters into some predefined categories for ease of interpretations or for supporting other data mining tasks. If the categories have existing data for training, this mapping can be recast into a standard text categorization problem, to which many solutions can be applied (Tseng & Juang, 2003; Yang & Liu, 1999).

Another need arises in that there is no suitable classification system at hand, but some generic labels are still desired for quick interpretations. For example, if documents in a cluster were talking about tables, chairs, and

Table 4
The confusion matrix of a term and a category

	Term T		
Category C		Yes	No
	Yes	TP	FN
	No	FP	TN

beds, then a title labeled ‘furniture’ would be perfect for this cluster, especially when this hypernym does not occur in this cluster. This case is often solved by human experts, such as those in Lai and Wu (2005), Glenisson, Glanzel, Janssens, and De Moor (2005), where cluster titles are given manually. Below we propose an automatic solution by use of an extra resource, i.e., WordNet.

WordNet is a digital lexical reference system developed by the Cognitive Science Laboratory at Princeton University (WordNet: a lexical database for the English language). English nouns, verbs, adjectives and adverbs are organized into synonym sets. Different relations, such as hypernym, hyponym, meronym, or holonym, are defined to link the synonym sets. With these structures, one can look up in WordNet all the hypernyms of a set of given terms and then choose the best among them with some heuristic rules. Since the hypernyms were organized hierarchically, the higher the level is, the more generic the hypernyms are. To maintain the specificity of the set of terms while revealing their general topics, the heuristics have to choose as low-level common hypernyms as possible. When there are multiple choices, ranks should be given to order the hypernyms in priority.

In our implementation, we look up for each given term all its hypernyms alone the path up to the root in the hierarchical tree. The number of occurrence (f) and the depth in the hierarchy (d) of an encountered hypernym are recorded. With the root being given a depth value of 0, a weight proportional to the normalized f and d is calculated for each hypernym as follows:

$$weight(hypernym) = \frac{f}{nt} \times 2 \times \left(\frac{1}{1 + \exp^{-c \times d}} - 0.5\right)$$

where nt is the number of given terms to normalize the occurrence f to a value ranges from 0 to 1 and c (0.125 in our implementation) is a constant to control the steepness of the sigmoid function $1/(1 + \exp(-c \times d))$ whose value approaches 1 (−1) for large positive (negative) d. Since the depth d only takes on non-negative value, the actual range of the sigmoid is from 0.5 to 1. It is thus subtracted with 0.5 and then multiplied by 2 to map the value of d into the normalized range: 0 to 1. Note that a term having no hypernym or not in WordNet is omitted from being counted in nt. Also note that a term can have multiple hypernyms and thus multiple paths to the root. A hypernym is counted only once for each given term, no matter how many times the multiple paths of this term pass this hypernym. This weight is finally used to sort the hypernyms in decreasing order to suggest priority.

Back to the previous example where the three terms were given: table, chair, and bed, their hypernym: “furniture” did result from the above calculation with a highest weight 0.3584 based on WordNet version 1.6.

3.7. Topic mapping

To represent the detected knowledge structures, two techniques are mainly used: HAC and MDS. Based on the pre-calculated similarities between each topic, the HAC method organizes the topics in a hierarchical way. This creates a structure that is readily available to the folder tree representation. In order to get higher intra-cluster similarities, the traditional complete-link clustering algorithm is chosen and developed as our HAC method. Similarly, from the pairwise similarities, the MDS technique computes the coordinates of each topic in specified dimensions of Euclidean space, which are usually 2 or 3 for ease of visual interpretation. With these coordinates, a topic map can be created by a plotting tool. We use the MDS program in the RuG/L04 package (Kleiweg (Software for Dialectometrics & Cartography)) for coordinate computation and the GD module in Perl for plotting. A circle is used to denote a topic. The size of the circle is designed to reflect the number of patents in it. Before plotting, the topics are further clustered by the complete-link algorithm with a specified threshold to divide them into several groups. Topics belonging to the same groups are plotted with the same colors so as to make the map more informative.

More advanced topic maps such as those introduced in Section 2 can be created by combining the structured information from the patents. For examples, by use of the patents from different time spans based on their filing or application dates, we may draw a series of growth maps or evolution maps. Specifically, patents not within the specified dates are removed from those clusters. The similarities among the resultant clusters are re-calculated and mapped based on the reduced number of patents. As long as the number of patents does not change largely, the change in the map should be minimal. With a series of such slightly changed

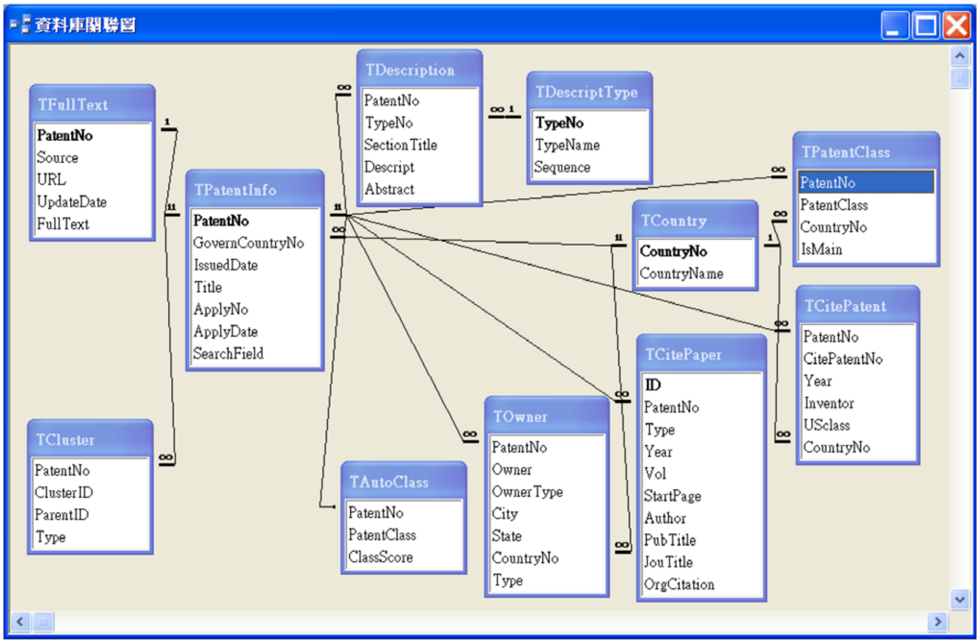


Fig. 5. The database schema for managing patents' structured information.

maps complied in an animated file format for visualization, one can trace the trend patterns in an intuitive way.

To facilitate such use of structured information, a database schema was designed. An example is shown in Fig. 5. Retrieving such information for use in various types of topic maps can be made easy and efficient through SQL queries.

4. Technique evaluation

This section presents the evaluation of important techniques discussed above. We believe that an effective text mining process relies on effective component techniques. Thus knowing their performance is important to gain confidence on the final text mining results.

Two patent collections were used in the evaluation: the major one is the 92 CNT patents introduced in the beginning; the other is the 612 patents from NSC (National Science Council), which will be discussed in more details in the next section.

4.1. Text segmentation

To evaluate the effectiveness of text segmentation and summarization, the full text of each patent was parsed into six segments from which a document set for each segment is created. Therefore, each set contains exactly the same number of documents as the original collection. Besides these six segment sets, we also created two other sets for later comparison: the segment extract set and the full-text set. Each is named with an abbreviation as follows:

1. **abs**: corresponds to the 'Abstract' section of each patent.
2. **abs**: corresponds to the segment of FIELD OF THE INVENTION.
3. **task**: corresponds to the BACKGROUND OF THE INVENTION.
4. **sum**: corresponds to the SUMMARY OF THE INVENTION.
5. **fea**: DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT.

6. **cla**: corresponds to the Claims section of each patent.
7. **seg_ext**: corresponds to the top-ranked sentences of each document from the sets: abs, app, task, sum, and fea.
8. **full**: corresponds to all the documents from the sets: abs, app, task, sum, and fea.

Each document in the segment extract set (seg_ext in the above) is the concatenation of the machine-generated summaries from each of the five segments (not including the Claims segment) in the same patent. The best *s* sentences are selected as the summary from each segment, where *s* is 6 in our implementation. Similarly, each document in the full set is the concatenation of the un-summarized texts from each of the five segments.

To show the segmentation results, among $6 \times 92 = 522$ segments in the CNT collection, only 9 do not contain any text, an empty rate of 1.63%. These include one empty document in the 'app' set, two in the 'task' set, five in the 'sum' set, and one in the 'fea' set. Among these nine empty documents, five are due to the lack of such sections in the original patent documents, three are due to the use of special section titles, and one is due to the erroneous spelling of the section title. The other two segments, the Abstract and Claims sections, always lead to correct extraction.

4.2. Text summarization

The use of segment summaries as document surrogates is an important feature in our approach. Knowing whether they are useful for subsequent processing and analysis is a key question to be answered. However, evaluation of automated summaries is not an easy task. Two main approaches are commonly applied: *intrinsic* and *extrinsic* (Mani, 2001). In intrinsic evaluation, manually prepared answers or evaluation criteria are compared with those which are machine generated. In extrinsic evaluation, automated summaries are evaluated based on their performance or influence on other tasks. We adopt the extrinsic approach since it is obviously suitable for our purpose.

In manual creation of a technology-effect matrix or a patent map for analysis, it is helpful to quickly spot the keywords that can be used for classifying the patents in the map. Once the keywords or *category features* are found, patents can usually be classified without reading all the texts. Thus a summary that retains as many important category features as possible is preferable. Our evaluation design therefore is to reveal whether the segment extract set (the seg_ext in the above) contains enough such features, compared to the other seven document sets.

In the following, the method for selecting category features for classification is introduced. By using the CNT patent map as our experimental data, the segments where important features occur are recorded for each patent and such occurrences are accumulated over all patents. The location distributions of important features among these segments are then compared.

4.2.1. Feature selection

Selecting the best category features for document categorization has been studied in the fields of machine learning and information retrieval. Yang and Pedersen (1997) compared five different methods. They found that Chi-square is among the best that lead to highest performance. The Chi-square method computes the relatedness of term *T* with respect to category *C* as:

$$\chi^2(T,C) = \frac{(\mathrm{TP} \times \mathrm{TN} - \mathrm{FN} \times \mathrm{FP})^2}{(\mathrm{TP} + \mathrm{FN})(\mathrm{FP} + \mathrm{TN})(\mathrm{TP} + \mathrm{FP})(\mathrm{FN} + \mathrm{TN})}$$

which is exactly the square of the correlation coefficient introduced previously. However, as pointed out by Ng et al. (1997), the correlation coefficient selects exactly those terms that are highly indicative of membership in a category, whereas the Chi-square method will not only pick out this set of terms but also those terms that are indicative of non-membership in that category. This is especially true when the selected terms are small in number. As an example, in a small real-world collection of 116 documents with only two exclusive categories: construction vs. non-construction in civil engineering tasks, some of the best and worst terms that are computed by Chi-square and correlation coefficient are shown in Table 5. As can be seen, due to the square nature,

Table 5
Some best and worst terms computed by Chi-square and correlation coefficient in a collection with two exclusive categories

Chi-square				Correlation coefficient			
Construction		Non-construction		Construction		Non-construction	
engineering	0.6210	**engineering**	0.6210	**engineering**	0.7880	equipment	0.2854
improvement	0.1004	improvement	0.1004	improvement	0.3169	procurement	0.2231
...				...			
kitchen	0.0009	kitchen	0.0009	communiqué	−0.2062	improvement	−0.3169
update	0.0006	update	0.0006	equipment	−0.2854	**engineering**	−0.7880

the chi-square weights negatively related terms as highly as positive ones. (For the term: ‘engineering’, the square of −0.7880 is 0.6210.) Therefore, instead of Chi-square, the correlation coefficient is used as our feature selection method.

4.2.2. Experiment results

In the technology-effect matrix of the CNT map, there are nine leaf-categories in the technology taxonomy and 21 leaf-categories in the effect taxonomy. Based on the correlation coefficient method, N ($N = 50$) top-ranked features for each of the leaf-category in each document set were extracted. The number of sets in which such a feature occurs, denoted *sc* for *set count*, was calculated. The features (less than $50 \times 8 = 400$) were then ranked by *sc* in descending order.

An example for some ranked features in the category FED (Field Emission Display) is shown in Table 6. The fifth row shows that the feature “electron” occurs in 5 document sets. It occurs in 27 documents in the ‘abs’ set and has a correlation coefficient of 0.31 in that set. The first column titled *rel* denotes whether the term is a relevant category feature or not. This is judged by one of the experts who participated in the creation of the CNT map. It should be noted that such a judgment is quite subjective and sometimes contradictive so that good features (such as “display” and “cathode” in this example) that are effective in discriminating the categories of the analyzed documents may not be judged as relevant. This is often observed in feature selection studies as statistically important terms are hard to identify manually. (For example, the three terms: “vs”, “cts”, and “loss” solely can achieve 93.5% classification accuracy for the largest category “earn” in the Rueters-21578 test collection (Bekkerman et al., 2001).) Also note that, from Table 6, the correlation coefficients in each segment set correlate to the set counts of the ordered features: the larger the set count, the larger the correlation coefficient in each segment set.

The M best features ranked by segment count were then selected, and are called *important category features* (ICFs). Their occurrences in each document set for each category were counted and averaged as:

$$\mathrm{MBTC}(s,c) = \frac{1}{M} \sum_{\mathrm{ICF} \in s \cap c} 1$$

where MBTC(s,c) stands for “M-Best Term Coverage of set s for category c”. Part of the results for $M = 30$ are shown in Table 7. As the first row shows (the FED category), among the 30 ICFs, 16 occur in the ‘abs’ set, covering 53% of them, and 21 in the ‘seg_ext’ set, covering 70% of the 30 terms, which reveals that most ICFs can be found in the best six sentences from each segment. The 30 ICFs were further checked by one of the

Table 6
Some feature terms and their distribution in each set for the category FED

Rel	term	sc	abs		app		task		sum		fea		cla		seg_ext		full	
yes	emission	8	20	0.69	17	0.59	31	0.62	21	0.73	34	0.63	20	0.63	33	0.64	40	0.54
yes	display	8	9	0.50	12	0.62	22	0.64	14	0.61	24	0.71	10	0.62	23	0.68	34	0.68
	cathode	8	12	0.39	9	0.42	27	0.48	14	0.54	30	0.53	15	0.51	25	0.52	41	0.47
	screen	5	2	0.27	2	0.27	8	0.37			18	0.43					19	0.41
yes	electron	5	27	0.31	25	0.40			36	0.28			27	0.37	61	0.35		
yes	voltage	4					20	0.45			45	0.37			16	0.28	52	0.39

Table 7
Occurrence distribution of 30 top-ranked terms in each set for some categories

Category	T_No	abs	app	task	sum	fea	cla	seg_ext	full
FED	30	16/53.3%	14/46.7%	22/73.3%	19/63.3%	21/70.0%	19/63.3%	21/70.0%	22/73.3%
Device	30	21/70.0%	17/56.7%	9/30.0%	16/53.3%	7/23.3%	19/63.3%	17/56.7%	8/26.7%
Electricity	30	12/40.0%	10/33.3%	10/33.3%	10/33.3%	8/26.7%	8/26.7%	13/43.3%	12/40.0%
Purity	30	12/40.0%	12/40.0%	7/23.3%	20/66.7%	9/30.0%	17/56.7%	18/60.0%	14/46.7%
Magnetic	30	18/60.0%	11/36.7%	6/20.0%	14/46.7%	14/46.7%	13/43.3%	15/50.0%	13/43.3%

Table 8
Occurrence distribution of manually judged terms in each set for some categories

Category	T_No	abs	app	task	sum	fea	cla	seg_ext	full
FED	7	6/85.7%	6/85.7%	6/85.7%	4/57.1%	6/85.7%	4/57.1%	6/85.7%	5/71.4%
Device	2	2/100.0%	1/50.0%	0/0.0%	1/50.0%	1/50.0%	2/100.0%	1/50.0%	0/0.0%
Electricity	2	2/100.0%	2/100.0%	0/0.0%	1/50.0%	1/50.0%	1/50.0%	0/0.0%	1/50.0%
Purity	8	6/75.0%	2/25.0%	3/37.5%	5/62.5%	1/12.5%	2/25.0%	4/50.0%	1/12.5%
Magnetic	2	2/100.0%	2/100.0%	1/50.0%	2/100.0%	1/50.0%	1/50.0%	2/100.0%	0/0.0%

CNT map creators. The results are shown in Table 8. The first row shows that among 7 relevant features, 6 of them occur in the 'abs' set, covering 87.5% of them.

The term-covering percentage (of the best terms and the relevant terms) of each set was accumulated and averaged over all categories with respect to the technology taxonomy and the effect taxonomy, respectively, as follows:

$$\mathrm{MBTC}(s) = \frac{1}{|\mathrm{Cat}|}\sum_{c\in\mathrm{Cat}}\frac{1}{M}\sum_{\mathrm{ICF}\in s\cap c} 1$$

where |Cat| denotes the number of categories in the taxonomy. The results are shown in Table 9. The second column denotes the number of categories (*nc*) in that taxonomy and the third column denotes the average number of terms in each category (*nt*) for calculating the term-covering average. The rows with a star in the first column denote that the average is calculated from the human-judged relevant terms. As the bold-faced data show, most machine-derived ICFs occur in the segment extracts, while most human-judged ICFs occur in the abstract section.

To see if important sets change when the number of ICFs changes, we varied the number M for additional values, i.e., 10 and 50, and calculated the averaged term-covering rates again. The results in Fig. 6 show that 'abs', 'sum' and 'seg_ext' are important sets. The 'full' set becomes important only when more terms are included for consideration.

4.2.3. *Findings*

From Fig. 6 and Table 9, it can be seen that: (1) Most ICFs ranked by correlation coefficient occur in the segment extracts, the Abstract section, and the SUMMARY OF THE INVENTION section. (2) Most ICFs selected by humans occur in the Abstract section or the Claims section. (3) The segment extracts lead to more

Table 9
Distribution of M-best term coverage in each segment averaged over all categories

Taxonomy	Set		abs	app	task	sum	fea	cla	seg_ext	full
	nc	nt								
Effect	9	$M = 30$	52.96%	41.48%	37.41%	53.33%	34.81%	47.04%	**57.04%**	41.85%
Effect*	8	4	**86.96%**	66.34%	45.40%	64.33%	51.03%	54.02%	55.09%	30.49%
Tech.	21	$M = 30$	49.37%	25.56%	26.51%	**56.51%**	34.44%	46.51%	**56.03%**	40.95%
Tech.*	17	4.5	**59.28%**	29.77%	23.66%	49.43%	34.46%	**60.87%**	44.64%	32.17%

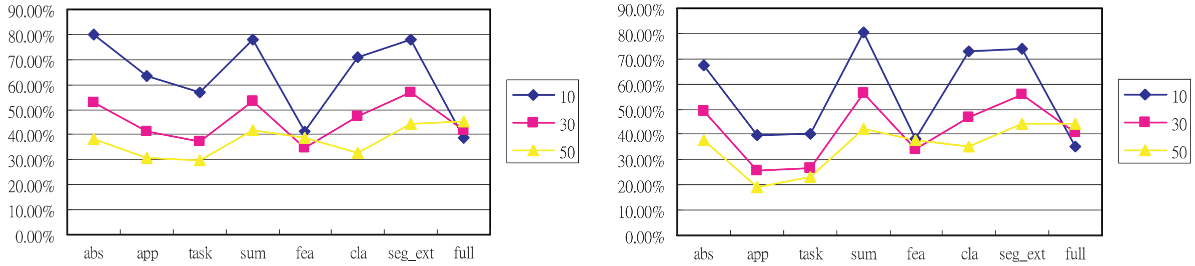


Fig. 6. Term-covering rates for M best terms, where $M = 10$, 30, and 50. The left figure is for the effect taxonomy and the right figure is for the technology taxonomy.

top-ranked ICFs than the full texts, regardless whether the category features are selected manually or automatically (Tseng, Juang, Wang, & Lin, 2005).

4.3. Key term extraction and association

Identifying multi-word phrases in an English document is somewhat like identifying words in a Chinese text, because both have no lexical delimiters. As such, we applied the key term extraction algorithm to the patent sets as well as some Chinese news documents to show its effectiveness.

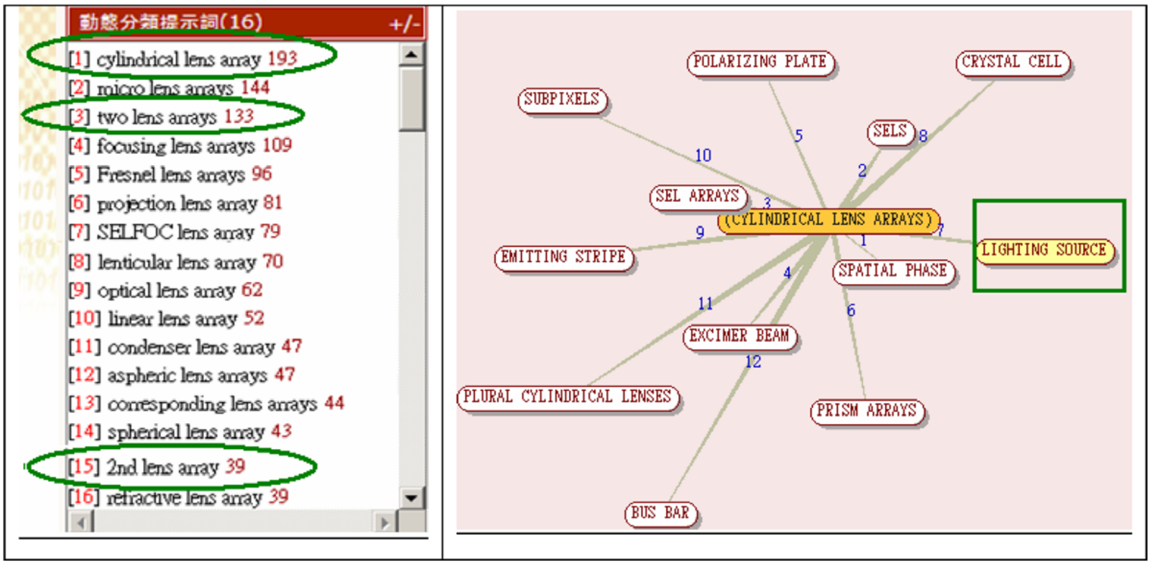
From the segment extract set (seg_ext) of the CNT patents, 942 multi-word phrases were extracted. They all occur at least twice in a document. Among them, only 132 occur in at least two documents. Most phrases are composed of two words (86 terms), and 31 terms are three-word phrases. An assessment of these 132 terms shows that there are 3 adjective phrases such as "efficient nonlinear", 15 short clauses such as "alter the porosity" and "resistance to fluid", and all others are correct noun phrases, yielding an error rate of $18/132 = 13.64\%$. To reduce the error rate, a stricter filtering rule such as removing those terms having stop words in them can be applied.

From a small collection of 100 Taiwan news articles, an average of 33 keywords (words that occur at least twice) in an article were extracted, in which an average of 11 terms (or 33%) of them were new to a lexicon of 123,226 terms. The total distinct 954 new terms contains 79 illegal words, an error rate of 8.3%. Compared to the total distinct 2197 extracted keywords, the error rate is only 3.6%. The longest new terms contain nine characters, while the shortest ones contain two characters.

The evaluation of the term association requires more manual efforts. To show the effectiveness in a large scale, a set of Chinese documents was used so that we can evaluate the quality of the co-words in a confident way.

In our experiment, 30 topics (single-term queries) were selected from the index terms of 25,230 Chinese news articles, from which term association was analyzed. Five assessors (all majored in library science) were invited for relevance judgment. For each topic, its top N ($N = 50$) co-words were examined. Users were asked if they thought the relationship between a topic and each of its co-words was relevant enough. If they are not sure (mostly due to lack of domain knowledge), they are advised in advance to retrieve those documents that may explain their associations. The results show that in terms of percentage of relatedness, 69% co-words were judged relevant to the topic terms in averege (Tseng, 2002). In another similar experiment with a much larger collection (154,720 documents), the percentage of relatedness increases to 78% (Ye, 2004). As can be seen, the more the documents for analysis, the better the effectiveness of the term association, a phenomenon that is commonly observed in robust machine learning algorithms.

The extracted keywords and their associated terms have a number of direct applications. They can be used in query expansion to improve retrieval performance (Tseng et al., 2004) or they can be suggested to analysts for prior art search or for exploring the knowledge structure underlying the collection. Fig. 7 demonstrates an application example, where a set of subsumed terms was suggested in response to the query "lens array" such that the sub-topics in the collection were revealed. Clicking on one of the suggested terms, a map would be displayed to show the relations of the term with other topics.


動態分類提示詞(16) +/-

[1] cylindrical lens array 193
[2] micro lens arrays 144
[3] two lens arrays 133
[4] focusing lens arrays 109
[5] Fresnel lens arrays 96
[6] projection lens array 81
[7] SELFOC lens array 79
[8] lenticular lens array 70
[9] optical lens array 62
[10] linear lens array 52
[11] condenser lens array 47
[12] aspheric lens arrays 47
[13] corresponding lens arrays 44
[14] spherical lens array 43
[15] 2nd lens array 39
[16] refractive lens array 39

Fig. 7. An application example of the extracted keywords and their associated terms.

4.4. Cluster title generation

For comparison, we used three methods to rank cluster terms for title generation, namely the modified Correlation Coefficient ($CC_{0.5}$), Term Frequency in Cluster (TFC), and multiplication of these two: CC × TFC. These ranking methods were applied to three sets of clustering results. The first is the first-stage document clustering from the CNT patents. The second is the second-stage document clustering from the CNT patents. The third is the third-stage term clustering from clustering the keywords extracted from NSC patents based on their co-words. For each cluster, at most 5 best terms were selected as its title. Two master students majored in library science then compared the relative quality of these terms under the same clusters. For each ranking method, the number of cases where it has the best title quality over the other methods was counted. Multiple best choices for a cluster are allowed and those cluster titles that are hard to assess can be omitted from being considered. The ranking methods are coded in such a way that the assessors do not know which method is used to generate the examined titles. The results are shown in Table 10. Note that hierarchical clustering structures were shown to the assessors. They were free to examine whatever clusters or sub-clusters they were interested in. This is why the numbers of examined clusters differ between them.

In spite of this difference, this preliminary experiment shows that titles generated by $CC_{0.5}$ or CC × TFC are favorable by one assessor, while those generated by TFC are not by either. This is somewhat surprised to know, since most past studies use TFC or a variation of it to generate cluster titles.

Table 10
Comparison of three title ranking methods among three cluster sets

Cluster set	Ranking method							
	Assessor	Number of clusters examined	Number of clusters that $CC_{0.5}$ is the best		Number of clusters that TFC is the best		Number of clusters that CC × TFC is the best	
First-stage	1	73	52	**71%**	5	7%	20	27%
doc. clusters	2	19	12	63%	6	32%	13	**68%**
Second-stage	1	13	9	**69%**	0	0%	6	46%
doc. clusters	2	7	3	43%	1	14%	5	**71%**
Third-stage	1	16	12	**75%**	5	31%	9	56%
term clusters	2	10	4	40%	5	50%	8	**80%**

Table 11
Cluster titles and their machine-derived categories

ID	Cluster's Title words	WordNet	InfoMap
1	resin, group, polymer, compound, methyl	1: substance, matter: 0.1853 **2: compound, chemical compound: 0.098** 3: whole: 0.0717	1: substance, matter: 2.472 2: object, physical object: 0.736 **3: compound, chemical compound: 0.5**
2	circuit, output, input, signal, voltage	**1: communication: 0.1470** **2: signal, signaling, sign: 0.1211** 3: production: 0.0823	**1: signal, signaling, sign: 1.250** **2: communication: 1.000** 3. round shape: 0.000
3	silicon, layer, material, substrate, powder	1: substance, matter: 0.1483 2: object, physical object: 0.1244 3: artifact, artefact: 0.1112	1: substance, matter: 2.250 2: artifact, artefact: 0.861 3: object, physical object: 0.833
4	system, edge, signal, type, device	1: artifact, artefact: 0.1483 **2: communication: 0.1470** 3: idea, thought: 0.0980	1: instrumentality: 1.250 **2: communication: 0.861** 3: artifact, artefact: 0.750
5	solution, polyaniline, derivative, acid, aqueous	1: communication: 0.1633 2: legal document,: 0.1540 3: calculation, computation: 0.1372	1: drug of abuse, street drug: 0.000 **2: compound, chemical compound: 0.0** 3: set: 0.000
6	sensor, magnetic, record, calcium, phosphate	1: device: 0.1514 2: object, physical object: 0.1244 **3: sound recording, audio recording: 0.12**	1: device: 0.312 2: fact: 0.000 3: evidence: 0.000
7	gene, cell, virus, infection, plant	1: structure, construction: 0.1225 2: contrivance, stratagem, dodge: 0.1155 3: compartment: 0.1029	1: entity, something: 0.790 **2: life form, organism, living thing: 0.5** 3: room: 0.000
8	density, treatment, strength, control, arrhythmia	1: property: 0.1112 2: economic policy: 0.1020 3: attribute: 0.0995	1: power, potency: 1.250 2: property: 0.674 3: condition, status: 0.625
9	force, bear, rod, plate, member	1: pistol, handgun, side arm: 0.1020 2: unit, social unit: 0.0980 3: instrumentality, instrumentation: 0.0980	1: unit, social unit: 1.250 2: causal agent, cause,: 0.625 3: organization: 0.500
10	transistor, layer, channel, amorphous, effect	1: artifact, artefact: 0.1390 2: structure, body structure: 0.1225 **3: semiconductor: 0.1029**	1: structure, anatomical structure: 0.500 2: artifact, artefact: 0.040 3: validity, validness: 0.000
1	acid, polymer, catalyst, ether, formula	1: substance, matter: 0.1853 2: drug: 0.0980 **3: compound, chemical compound: 0.098**	**1: compound, chemical compound: 1.25** 2: substance, matter: 1.062 3: object, physical object: 0.484
2	silicon, layer, transistor, gate, substrate	1: object, physical object: 0.1244 2: device: 0.1211 3: artifact, artefact: 0.1112	1: object, physical object: 0.528 2: substance, matter: 0.500 3: region, part: 0.361
3	plastic, mechanism, plate, rotate, force	1: device: 0.1514 2: base, bag: 0.1155 3: cut of beef: 0.1155	1: device: 0.361 2: entity, something: 0.236 3: chemical process, chemical action: 0.0
4	output, signal, circuit, input, frequency	**1: communication: 0.1470** **2: signal, signaling, sign: 0.1211** 3: relation: 0.0995	**1: signal, signaling, sign: 1.250** **2: communication: 1.000** 3: abstraction: 0.268
5	powder, nickel, electrolyte, steel, composite	1: substance, matter: 0.1483 **2: metallic element, metal: 0.1211** 3: instrumentality, instrumentation: 0.0980	**1: metallic element, metal: 0.500** 2: substance, matter: 0.333 3: entity, something: 0.203
6	gene, protein, cell, acid, expression	1: substance, matter: 0.1112 2: object, physical object: 0.0995 3: compound, chemical compound: 0.0980	1: entity, something: 0.893 2: compound, chemical compound: 0.500 3: object, physical object: 0.026

4.5. Mapping cluster titles to categories

The proposed title mapping algorithm is applied to two real-world cluster sets: the first is a term cluster set, the second is a document cluster set, both are the final-stage clustering results from the NSC patents to be discussed in the next section. The first set has 10 clusters and the second has 6. Their cluster titles are shown in the second column of Table 11.

The proposed method is compared to a similar tool called InfoMap (Information Mapping Project) which is developed by the Computational Semantics Laboratory at Stanford University. This online tool finds a set of taxonomic classes for a list of given words. It seems that WordNet is also used as its reference system, since the output classes are mostly WordNet's terms. An agent program is written to send the title words to InfoMap and collect the results that it returns. Only the top-three candidates from both methods are compared. They are listed in the last two columns in Table 11, with their weights appended.

The reasonable classes are marked in boldface in the table. As can be seen, the two methods perform similarly. Both achieve a level of 50% accuracy in either set.

5. Application example

5.1. The NSC patent set

One of the objectives of this work is to help analyze the US patents whose assignee is National Science Council (NSC). NSC is the major government agency that sponsors research activities in Taiwan. Institutes, universities, or research centers, public or private, can apply for research fundings from NSC. Once the research results yield any US patents, the intellectual property rights belong to NSC. In other words, NSC becomes the assignee of the patents. However, this policy has been changed since year 2000. NSC no longer insisted on the ownership of the rights. The applicants own the management rights of these intellectual properties.

Due to this background, these documents constitute a knowledge-diversified collection with relatively long texts (about 2000 words per document) describing various advanced technical details. Analysis of the topics in this collection becomes a non-trivial task as very few analysts know of such diversified technologies and fields. Although each patent has pre-assigned International Patent Classification (IPC) or US Patent Classification (UPC) codes, many of these labels are either too general or too specific to fit the intended knowledge structures for topic interpretation. Therefore, using them alone does not meet the requirement of the analysis. As such, organizing the collection by text mining techniques becomes an important method to help humans understand this collection.

The NSC collection was created by searching the USPTO's web site using "National Science Council" as the search term limited to the assignee field. A total of 612 patents were downloaded on 2005/06/15.

5.2. Text mining processing

The 612 patents were parsed, segmented, and summarized. Among the $6 * 612 = 3672$ segments, only 79 empty segments resulted, yielding an empty rate of 2.15%. The full texts of these patents take up 23.9 MB. After summarization with at most 6 sentences for each of the 5 segments (the Claims segment is excluded from the task of topic analysis), the size becomes 2.93 MB, a compression ratio of 87.74%. In spite of this, the lost important terms are few. From the full text set, a total of 20,072 keywords (terms occur at least twice in a document) were extracted. Among them, 19,343 can be found from the 5-segment summary set. Most of the 20,072 keywords occur in one document. Only 2714 of them occur in at least two and have at least one co-occurred terms associated with them.

These 2714 terms were then used to index the document surrogates. With these concise representations in documents and terms, a set of topic maps were generated.

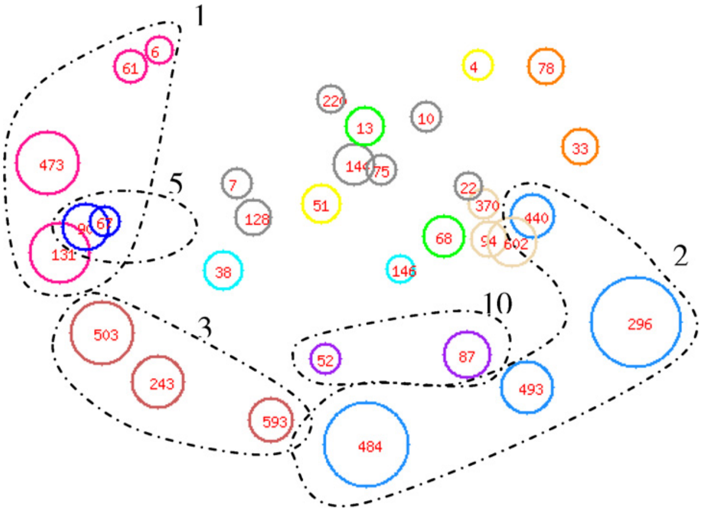


Fig. 8. Topic map based on term clustering from the NSC patents.

5.3. Topic mapping

The terms and their co-occurred words alone can be used to generate a topic map for the collection. The above 2714 terms were clustered by a complete-link method, based on how many co-occurred terms they share, into 353 small-size clusters. These clusters were then repeatedly clustered, again based on the common co-occurred terms, into 101 medium-size clusters, then into 33 large-size clusters, and finally into 10 topic clusters. During this multi-stage clustering, the threshold, the *low_tf* to filter out those terms whose $TF <= low\_tf$, is gradually set to a higher value (from 1, 2, 4, to 6) to reflect the need for the clusters to be more and more general in topics in each successive stage. The reason that we performed this multi-stage clustering is due to the fact that even we set a lowest threshold, we could not obtain as low as 10 clusters in a single step for reasonable visual interpretation. The 353 small-size clusters are actually obtained with the similarity threshold set to 0.0. In other words, among the 2714 terms, no two terms in different small-size clusters share the same co-words.

The 33 large-size clusters were mapped by the MDS method as shown in Fig. 8, in which clusters belonging to the same topics were painted with the same colors. Note the clusters in grey color were not grouped with the others, they form individual topics on their own, not counted in the above 10 topics. Table 12 lists part of these clusters and their topic titles. Topic 1 includes clusters with IDs: 131, 473, 6, and 61, which correspond to the red circles[1] located in the left-upper corner in the map. As can be seen from the title words, this topic is about chemical compounds, which can be generated from our WordNet-lookup algorithm automatically. Other cluster's categories are shown in Table 11, where only 50% of them are reasonable. If manual labeling is allowed, some topics can be easily labeled from the cluster titles. An example is topic 7, where "biology" seems to be a reasonable category name from its title words.

Note the above clustering did not involve any information from the documents themselves, except for the initial stage where the full texts are used to extract key terms and their co-words. As another way for topic analysis, the 612 NSC patents were clustered based on their summary surrogates. They were first clustered into 91 topics, which in turn were grouped in 21 sub-domains, from which 6 major domains were found. Fig. 9 shows the relative closeness among these 21 sub-domains. Table 13 lists the hierarchical results. Again, their machine-derived cluster categories are shown in Table 11, with 50% accuracy. For manual labeling, we tagged each major domain in Fig. 9 by reading their title words in Table 13 based on the academic divisions to be discussed later. With some basic knowledge in science and technology, each domain was labeled without difficulty, except domain 3 whose title words are more diversified. It was thus labeled Generality for general topics.

[1] For interpretation of the references in colour in this figure, the reader is referred to the Web version of this article.

Table 12
Final-stage term clusters from the NSC patents

```
 1: 313 terms: 0.110708 (resin: 67.0, group: 50.3, polymer: 49.8, compound: 33.8, methyl: 31.3)
    o 5: 247 terms: 0.517729 (resin: 99.0, group: 83.3, polymer: 77.5, acid: 47.3, formula: 46.0)
      + ID = 131: 120 terms: 0.262248(resin: 48.2, alcohol: 36.0, polymer: 28.6, group: 24.3, phenolic: 21.0)
      + ID = 473: 127 terms: 0.130294(phenyl: 34.7, amino: 21.0, acid: 20.9, formula: 20.3, activity: 12.3)
    o 11: 66 terms: 0.425685 (roller: 31.9, cam: 22.2, compound: 20.7, pitch: 18.0, thread: 14.5)
      + ID = 6: 29 terms: 0.6842(compound: 15.2, methyl: 5.9, tube: 5.6, borohydride: 5.6, phosphonium: 4.9)
      + ID = 61: 37 terms: 0.343894(roller: 30.9, cam: 21.8, variable: 18.0, pitch: 17.6, mechanic: 16.1)

 2: 655 terms: 0.193269 (circuit: 249.9, output: 243.0, input: 223.0, signal: 194.3, voltage: 162.5)
   * 3: 592 terms: 0.526790 (signal: 238.3, output: 199.9, layer: 198.7, input: 182.9, light: 177.0)
    o 1: 500 terms: 0.708007 (circuit: 209.4, signal: 159.7, output: 151.6, layer: 140.6, input: 138.8)
      + ID = 484: 236 terms: 0.1288(layer: 138.8, substrate: 55.2, semiconductor: 47.2, schottky: 41.3, ...)
      + ID = 296: 264 terms: 0.177278(output: 148.5, circuit: 107.8, signal: 102.3, input: 61.7, terminal: 53.8)
    o ID = 493: 92 terms: 0.126080(optic: 46.6, light: 35.3, field: 17.5, index: 15.7, layer: 13.1)
   * ID = 440: 63 terms: 0.136660(circuit: 20.7, line: 9.5, voltage: 4.0)

 3: 289 terms: 0.361452 (silicon: 49.4, layer: 41.6, material: 40.2, substrate: 33.8, powder: 33.4)
   * 13: 220 terms: 0.399435 (dielectric: 40.0, ceramic: 34.6, layer: 29.2, silicon: 28.5, catalyst: 28.4)
    o ID = 243: 86 terms: 0.200064(dioxide: 24.2, liquidate: 19.8, phase: 19.0, substrate: 10.9, deposit: 9.4)
    o ID = 503: 134 terms: 0.1236(ceramic: 49.0, material: 17.5, dielectric: 17.0, powder: 16.4, mixture: 15.7)
   * ID = 593: 69 terms: 0.102624(thermal: 15.7, silicon: 11.5, film: 4.7, substrate: 2.6)

 4: 159 terms: 0.186075 (system: 38.5, edge: 32.0, signal: 13.5, type: 7.4, device: 6.8)
   * 35: 129 terms: 0.289348 (edge: 40.0, flow: 32.6, system: 26.0, vortex: 25.6, air: 23.6)
    o ID = 94: 47 terms: 0.304667(air: 23.2, edge: 14.2, upstream: 13.4, transfer: 12.0, system: 11.1)
    o ID = 602: 82 terms: 0.100589(flow: 25.8, vortex: 23.8, pressure: 19.5, fluid: 18.9, pipe: 16.8)
   * ID = 370: 30 terms: 0.153259(signal: 8.5, fourier: 7.0, demodulator: 6.3, software: 6.3, value: 6.3)

 5: 105 terms: 0.365060 (solution: 34.2, polyaniline: 20.1, derivative: 14.5, acid: 9.5, aqueous: 9.5)
   * ID = 90: 71 terms: 0.305932 (solution: 23.4, copper: 7.7, pitch: 7.0, sieve: 7.0, molecular: 7.0)
   * ID = 67: 34 terms: 0.335136 (polyaniline: 19.7, derive: 13.8, derivative: 13.4, water: 7.0, solvent: 7.0)

 6: 77 terms: 0.250290 (sensor: 16.9, magnetic: 14.2, record: 10.4, calcium: 9.0, phosphate: 9.0)
   * ID = 38: 51 terms: 0.392179 (magnetic: 14.2, record: 9.1, calcium: 8.4, composition: 8.2, phosphate: 7.7)
   * ID = 146: 26 terms: 0.252356 (sensor: 7.2, mndr: 5.6, fabrication: 4.9, temperature: 4.2, ethanol: 4.2)

 7: 109 terms: 0.353365 (gene: 31.9, cell: 26.4, viru: 16.6, infection: 13.9, plant: 13.9)
   * ID = 68: 57 terms: 0.332308 (combustion: 9.1, chamber: 9.1, cell: 7.0, nip: 7.0, solar: 7.0)
   * ID = 13: 52 terms: 0.496672 (gene: 45.0, viru: 18.7, infection: 15.4, plant: 13.8, sugar: 8.9)

 8: 84 terms: 0.256320 (density: 16.6, treatment: 12.5, strength: 10.4, control: 9.1, arrhythmia: 8.3)
   * ID = 51: 53 terms: 0.359096 (density: 23.0, film: 2.3)
   * ID = 4: 31 terms: 0.782113 (treatment: 8.3, arrhythmia: 7.7, hypertension: 5.6, display: 5.6, breast: 5.6)

 9: 86 terms: 0.505293 (force: 42.0, bear: 28.0, rod: 20.1, plate: 18.4, member: 15.6)
   * ID = 78: 43 terms: 0.316616 (force: 24.4, bear: 11.2, steel: 10.6, application: 10.5, pawl: 8.4)
   * ID = 33: 43 terms: 0.407034 (member: 23.1, rod: 19.0, end: 15.4, plate: 10.1, connect: 9.8)

10: 104 terms: 0.300222 (transistor: 29.8, layer: 27.4, channel: 20.1, amorphous: 15.9, effect: 11.8)
   * ID = 52: 34 terms: 0.358806 (amorphous: 14.1, amorphous silicon: 8.4, design: 4.9, uhv: 4.2, ...)
   * ID = 87: 70 terms: 0.306677 (effect: 23.1, rat: 9.1, transistor: 9.1, ingaa: 8.4, region: 7.7)
```

More types of advanced analysis can be obtained by combining the topic maps with the patents' structured information. Their implementation involves techniques from the fields of database management and user interface. The details of which are beyond the scope of this discussion.

5.4. Topic analysis comparison

The topic distribution of the NSC patents had actually been analyzed by a subsidiary center of NSC: Science and Technology Information Center (STIC). STIC analyzed this patent set first by IPC. The results in the form of the topic tree are shown in Table 14, where major categories are shown in boldface and with their IPC

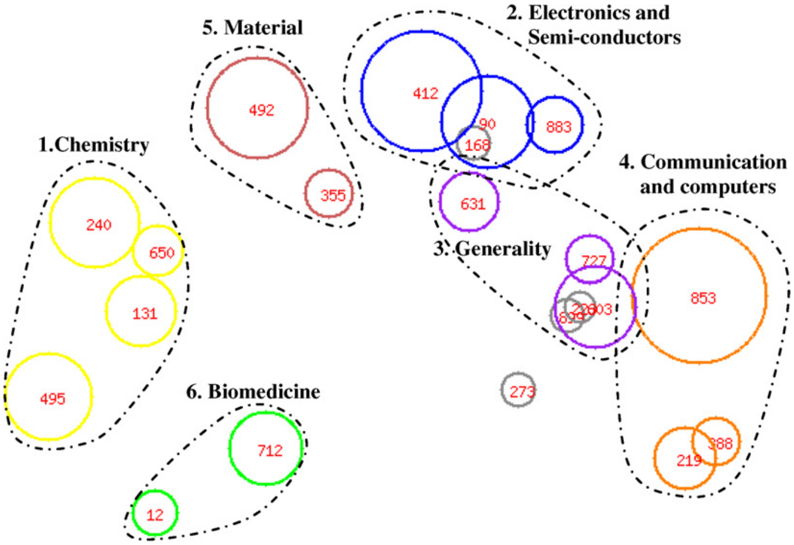


Fig. 9. Topic map based on document clustering from the NSC patents.

Table 13
Final-stage document clusters from the NSC patents

1: 122 docs.: 0.201343 (acid: 174.2, polymer: 166.8, catalyst: 155.5, ether: 142.0, formula: 135.9)
  * 108 docs.: 0.420259 (polymer: 226.9, acid: 135.7, alkyl: 125.2, ether: 115.2, formula: 110.7)
    o 69 docs.: 0.511594 (resin: 221.0, polymer: 177.0, epoxy: 175.3, epoxy resin: 162.9, acid: 96.7)
      + ID = 131: 26 docs.: 0.221130(polymer: 86.1, polyimide: 81.1, aromatic: 45.9, bis: 45.1, ether: 44.8)
      + ID = 240: 43 docs.: 0.189561(resin: 329.8, acid: 69.9, group: 57.5, polymer: 55.8, monomer: 44.0)
    o ID = 495: 39 docs.: 0.138487(compound: 38.1, alkyl: 37.5, agent: 36.9, derivative: 33.6, formula: 24.6)
  * ID = 650: 14 docs.: 0.123005(catalyst: 88.3, sulfide: 53.6, iron: 21.2, magnesium: 13.7, selective: 13.1)

2: 140 docs.: 0.406841 (silicon: 521.4, layer: 452.1, transistor: 301.2, gate: 250.1, substrate: 248.5)
  * 123 docs.: 0.597062 (silicon: 402.8, layer: 343.4, transistor: 224.6, gate: 194.8, schottky: 186.0)
    o ID = 412: 77 docs.: 0.150265(layer: 327.6, silicon: 271.5, substrate: 178.8, oxide: 164.5, gate: 153.1)
    o ID = 90: 46 docs.: 0.2556 (layer: 147.1, schottky: 125.7, barrier: 89.6, heterojunction: 89.0, transistor: ...)
  * ID = 883: 17 docs.: 0.103526 (film: 73.1, ferroelectric: 69.3, thin film: 48.5, sensor: 27.0, capacitor: 26.1)

3: 66 docs.: 0.220373 (plastic: 107.1, mechanism: 83.5, plate: 79.4, rotate: 74.9, force: 73.0)
  * 54 docs.: 0.308607 (plastic: 142.0, rotate: 104.7, rod: 91.0, screw: 85.0, roller: 80.8)
    o ID = 631: 19 docs.: 0.125293 (electromagnetic: 32.0, inclin: 20.0, fuel: 17.0, molten: 14.8, side: 14.8)
    o ID = 603: 35 docs.: 0.127451 (rotate: 100.0, gear: 95.1, bear: 80.0, member: 77.4, shaft: 75.4)
  * ID = 727: 12 docs.: 0.115536 (plasma: 26.6, wave: 22.3, measur: 13.3, pid: 13.0, frequency: 11.8)

4: 126 docs.: 0.457206 (output: 438.7, signal: 415.5, circuit: 357.9, input: 336.0, frequency: 277.0)
  * 113 docs.: 0.488623 (signal: 314.0, output: 286.8, circuit: 259.7, input: 225.5, frequency: 187.9)
    o ID = 853: 92 docs.: 0.105213 (signal: 386.8, output: 290.8, circuit: 249.8, input: 224.7, light: 209.7)
    o ID = 219: 21 docs.: 0.193448 (finite: 41.3, data: 40.7, architecture: 38.8, computation: 37.9, algorithm: ...)
  * ID = 388: 13 docs.: 0.153112 (register: 38.9, output: 37.1, logic: 32.2, address: 28.4, input: 26.2)

5: 64 docs.: 0.313064 (powder: 152.3, nickel: 78.7, electrolyte: 74.7, steel: 68.6, composite: 64.7)
  * ID = 355: 12 docs.: 0.1586 (polymeric electrolyte: 41.5, electroconductive: 36.5, battery: 36.1, electrode: ...)
  * ID = 492: 52 docs.: 0.138822 (powder: 233.3, ceramic: 137.8, sinter: 98.8, aluminum: 88.7, alloy: 63.2)

6: 40 docs.: 0.250131 (gene: 134.9, protein: 77.0, cell: 70.3, acid: 65.1, expression: 60.9)
  * ID = 12: 11 docs.: 0.391875 (vessel: 30.0, blood: 25.8, platelet: 25.4, dicentrine: 17.6, inhibit: 16.1)
  * ID = 712: 29 docs.: 0.116279 (gene: 148.3, dna: 66.5, cell: 65.5, sequence: 65.1, acid: 62.5)

descriptions. Those minor categories are listed together with their numbers of patents shown in the parentheses. As can be seen, these patents spread over a large range of IPC categories and many of such categories are

Table 14
Breakdown of major IPC categories for the NSC patents

A: 87 docs.: Human Necessities
+ **A61**: 71 docs.: Medical Or Veterinary Science; Hygiene
+ A*: 16 docs.: A01(7), A21(2), A23(2), A42(2), A03(1), A62(1), A63(1)

B: 120 docs.: Performing Operations; Transporting
+ **B01**: 25 docs.: Physical Or Chemical Processes Or Apparatus In General
+ **B05**: 28 docs.: Spraying Or Atomising In General; Applying Liquids Or Other Fluent Materials To Surfaces
+ **B22**: 17 docs.: Casting; Powder Metallurgy
+ B*: 50 docs.: B32(12), B29(11), B62(6), B23(4), B24(4), B60(4), B02(2), B21(2), B06(1), B25(1), ...

C: 314 docs.: Chemistry; Metallurgy
+ **C07**: 62 docs.: Organic Chemistry
+ **C08**: 78 docs.: Organic Macromolecular Compounds; Their Preparation Or Chemical Working-Up; ...
+ **C12**: 76 docs.: Biochemistry; Beer; Wine; Vinegar; Microbiology; Mutation Or Genetic Engineering; ...
+ C*: 98 docs.: C23(22), C25(20), C01(19), C04(10), C09(10), C22(8), C03(5), C30(3), C21(1)

D: 6 docs.: Textiles; Paper

E: 8 docs.: Fixed Constructions

F: 30 docs.: Mechanical Engineering; Lighting; Heating;

G: 134 docs.: Physics
+ **G01**: 49 docs.: Measuring; Testing
+ **G02**: 28 docs.: Optics
+ **G06**: 29 docs.: Computing; Calculating; Counting
+ G*: 28 docs.: G10(7), G11(7), G05(6), G03(5), G08(2), G09(1)

H: 305 docs.: Electricity
+ H01: 216 docs.: Basic Electric Elements
    o **H01L021**: 92 docs.: Processes or apparatus adapted for the manufacture or treatment of semiconductor
    o **H01L029**: 35 docs.: Semiconductor devices adapted for rectifying, amplifying, oscillating, or switching;
    o H01L*: 89 docs.: others.
    O H01*: 53 docs.: H03K(23), H03M(11), H04B(10), H04L(7), H04N(7), H01B(5), H03H(4), H04J(4), ...
+ **H03**: 51 docs.: Basic Electronic Circuitry
+ **H04**: 30 docs.: Electric Communication Technique
+ H*: 8 docs.: H05(5), H02(3)

either ambiguous or not at the detailed-enough level or abstract-enough level that is needed by an analyst. Further breakdown of those ambiguous categories yields similarly divergent and skewed distribution. This makes them hard for further analysis.

As an alternative, STIC used the division information of these patents to show their topic distribution, as is shown in Table 15. This academic division information comes from the actual divisions of NSC where the

Table 15
Division distributions of the NSC patents

Abbrev.	Division	Percentage
Ele	Electrical Engineering	28.63
Che	Chemical Engineering	14.70
Mat	Material Engineering	14.12
Opt	Optio-Electronics	13.15
Med	Medical Engineering	10.44
Mec	Mechanical Engineering	6.58
Bio	Biotechnology Engineering	5.03
Com	Communication Engineering	2.90
Inf	Information Engineering	2.90
Civ	Civil Engineering	1.16
Others		0.39
Total		100.00

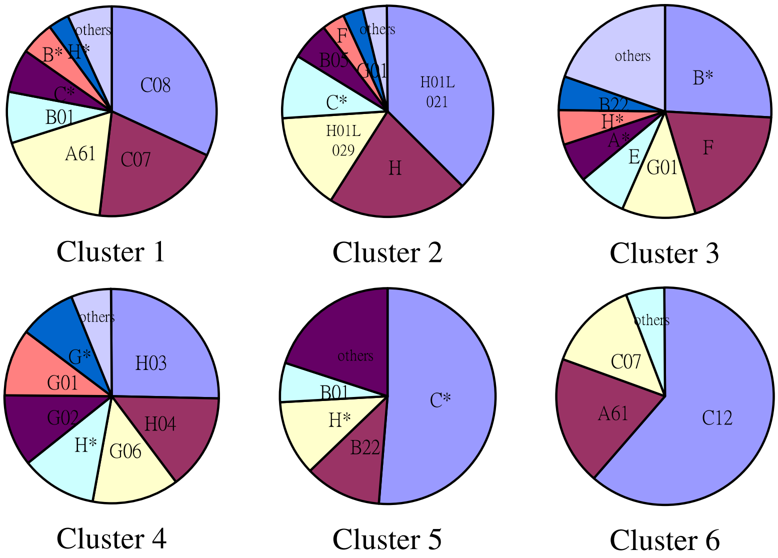


Fig. 10. Distribution of major IPC categories in each cluster.

patents were managed. In each division, the patent trend (distribution of numbers of patents over years) and the top-performing institutes (those apply for most patents) were then analyzed by STIC, where the institute information again comes from NSC.

The division-wise analysis is quite independent of each other. As more interactions are involved in nowadays researches, inter-disciplinary relations are interesting to monitor. To reflect this need, the text mining approach was applied, because it relates patents not only by predefined classification, but also by the content they share. As Fig. 9 shows the relationship among the clusters, Figs. 10 and 11 further reveal the IPC and division distributions in each cluster, respectively.

In Fig. 10, the IPC categories now become disambiguated when considering the other co-occurred categories in the same cluster. For example, in Cluster 1, A61 co-occurred with C08 and C07 such that the patents in

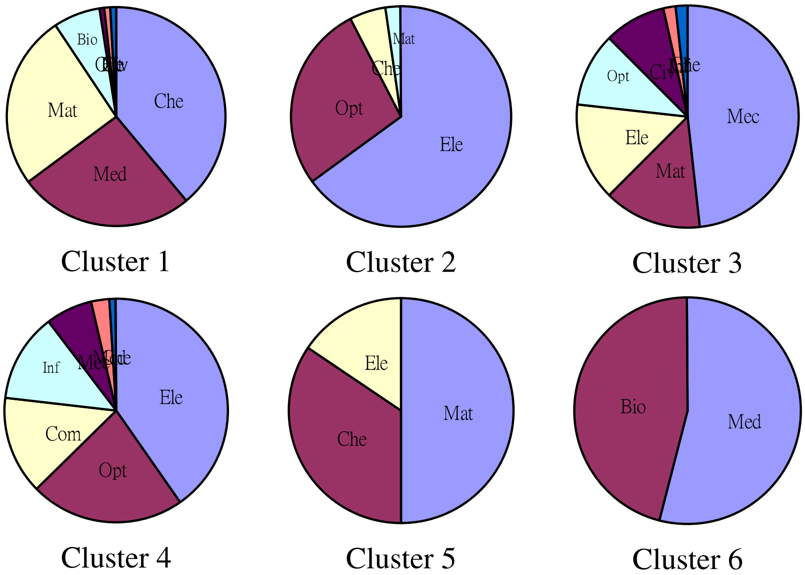


Fig. 11. Distribution of academic divisions in each cluster.

this cluster should be more about medical chemistry, while less about hygiene. In Cluster 6 it co-occurred with C12 such that the patents should be more about biomedical chemistry or biomedicine, while less about beer, wine, etc. Fig. 11 supports these observations in all these six clusters.

Comparing among Fig. 9, Tables 14 and 15, these three classification systems provide different facets to understand the topic distribution of the patent set. Each may reveal some insights if we can interpret it. The IPC system results in divergent and skewed distributions which make it hard for further analysis (such as trend analysis). The division classification is the most familiar one to the NSC analysts, but it lacks inter-disciplinary information. As to the text mining approach, it dynamically glues related IPC categories together based on the patent contents to disambiguate their vagueness. This makes future analysis possible even when the division information is absent, as may be the case in later published patents to which NSC no longer claims their right.

6. Discussions

This section discusses the findings, experiences, and implications from the above application examples and evaluation results. Related studies are also reviewed, of which some inspire our solutions and some compensate our work.

The patent sections appear quite regular and each has its own functions in describing the innovations. This regularity makes the segment matcher a simple case to design. Our ad hoc implementation shows only a fraction of segments (less than 3%) fails to be extracted. Most failure cases are due to the omission of the corresponding sections.

The purpose of summarization is to facilitate people's understanding of a text quickly or easily. In the patent analysis scenario for patent map generation, the tasks include quickly spotting the topic-relevant sections and in those spotted sections quickly focusing on the topic-indicative terms for classification. Our work through document segmentation and summarization has provided strong evidence to help solve these tasks. Specifically, our results show that the Abstract, Summary, and summaries from each section in a patent document are the most topic-relevant sections, regardless of whether the classification is for technological aspects or functional aspects. The implication is that if one would like to quickly determine a patent's category based on only a few terms in a quick pace, one should first read the Abstract section or the Summary section. Or alternatively, one could first browse the "segment extracts" generated automatically.

However, our heuristic-based method needs further improvement to yield summaries suitable for direct manual analysis, especially for the "summary of the invention" and "detailed description" sections. Although "problems to be solved" extracted from the background section seem feasible, as shown in our example, automated summarization of "the solution methods" is still a challenge. In addition, the claims in patent documents are written in a special style that is not easy to read. It is thus also an important part that needs further processing. Recent studies have shown promising in transforming, although not summarizing, original long claims into short sentences to increase readability (Sheremetyeva, 2003; Shinmori, Okumura, Marukawa, & Iwayama, 2003).

An efficient key term extraction algorithm is presented and used in the text mining process. Although Fagan (1989) and Smadja (1993) had presented similar phrase extraction algorithms without using machine-readable dictionaries, their methods use corpus-wide statistical information such as document frequency of a term or a pair of terms. As such, their results and effectiveness depend on the corpus used. In contrast, ours can be considered as rule-based. It uses only three simple rules to respectively drop, merge, and accept the sub-phrase units during a merging back process based only on their term frequencies within a document. The accepted repeated patterns are then filtered by a stopword list. As such, its effectiveness depends on whether our assumption (i.e., "If a document focuses on a topic, it is likely to mention a set of strings repeatedly") holds for the document language and the domain we are dealing with. For languages like Chinese or for domains like the technical patents, our method works well in terms of non-dictionary topical terms that can be extracted.

The co-word analyses based on document-wide co-occurrence or latent semantic indexing require enormous computation for large collections. Therefore, an analysis based on snippet-wide co-occurrence is devised. Although such an idea has been proposed previously, such as those in Brown, Della Pietra, De Souza, Mercer, and Lai (1992) and Schutze (1993), the difference between ours and others lies in the refinement. As our experiments showed (Tseng, 2002), a naïve implementation of the idea leads to only 48% of co-occurred terms judged as relevant, while a refined implementation leads to 69%. It is interesting to note that, as with many robust machine learning algorithms, the more the documents for analysis, the better the results from our association computation. Furthermore, through the clustering of terms based on their co-words, terms not occurring in the same documents can be associated.

The term suggestion mechanism based on the extracted keywords and their co-words has been mentioned by Larkey (1999). At the request of the US patent office, Larkey provided such a search aid with the Inquery system which uses WordNet to help extract phrases and uses section-level co-occurrence to extract co-words. As Fujii et al. (2004) showed in the task of invalidity search, the machine-generated results from 30 systems do not cover all the relevant patents, manual searches contribute as many relevant ones. These evidences show that hybrid approaches like term suggestion would be of great help to prior art search.

A multi-stage clustering approach is used to mine the knowledge structures and transform them from concepts to topics and in turn from topics to categories. Three ranking functions to select the cluster titles are compared. Preliminary evaluation shows that the ranking method commonly used in past studies does not yield favorable results when compared to the others.

Cluster labeling is important for ease of human analysis. Uchida, Mano, and Yukawa (2004) showed that although their clusters coincided with human's for a high percentage, their method failed to produce good cluster labels because human labels are mostly compound nouns, while theirs are mostly single words due to the difficulties of determining compound words. Their work reminded us the importance of phrase extraction and motivated us to map the cluster titles to more generic labels.

In the attempt to produce generic cluster labels, a hypernym search algorithm based on WordNet is devised. Real-case evaluation shows that only 50% cluster labels lead to reasonable categories. However, this is due to the limit of WordNet in this case. WordNet does not cover all the terms extracted from the NSC patents. Also WordNet's hypernym structures may not reflect the knowledge domains desired for analyzing the NSC patents. If a suitable classification system can be found, the hypernym search algorithm may lead to more desirable results. Because the algorithm uses only the depth and occurrence information of the hypernyms, it is general enough to be applicable to any hierarchical systems.

The clustering results are plotted on 2-dimensional topic maps with the MDS method. This way of visualizing the results has been used in scientometrics for technology watch or scientific policy decision. However, insights are still hard to derive from these maps alone. After years of mapping scientific domains, Noyons and Buter commented that such an overview (via the visualized maps) can be used to find information if you are able to interpret it. In a science and technology foresight analysis based on publication citations (The 8th Science & Technology Foresight Survey, 2004), questionnaires were gathered from 37 experts in 38 domains for comments and feedback. Some experts confirmed that these maps and topics analyses are valuable, whereas some could not see the practical use of them because of their uninterpretability or improper labels and organization of the topical domains.

In our case, the topic maps created from the NSC patents were presented to three analysts for their comments. One common reaction is that without reading any patents, they are able to have a multilayered overview of them, which is a great relief. Although from IPC or UPC codes, similar overviews can be generated, as has been shown, these pre-defined categories do not meet the domain structures desired by the analysts. A second reaction is that if trend types can be classified and marked on the maps for each cluster, spotting and tracing the trend patterns on the maps can be easier. The suggested types include quantity-based trends such as changing ratios of number of patents in a cluster and quality-based trends such as changing ratios of number of citations to the patents in a cluster. Another valuable comment suggests replacing each circle on the map with a pie chart in order to show the distribution of different assignees or countries for comparative studies and competitive analyses. This would allow them to spot which domains are dominated by which institutes or countries, for example. Analytical information like this could be very useful for policy making. In short, by combining structured information in a friendly interface for various types of interactive analysis and visualization, the topic maps could lead to valuable insights for analysts.

In recent years, SOM has also been applied to patent analysis. The variation WEBSOM method has been tested on nearly 7 millions of patent abstracts (Lagus, Kaski, & Kohonen, 2004). Nevertheless, Lamirel, Shadi

Al Shehabi, Hoffmann, and Francois (2003) pointed out that this method only provides the analyst with general overview of the topics covered by the patents. They then proposed a MultiSOM model that introduces the concepts of viewpoints and dynamics to assist information analysis based on its multi-map display and its inter-map communication process. Specifically, each viewpoint corresponds to a map which is created based on a specific section of the patent. Inter-map communications are activated through the same set of patents in different maps. The advantages of the MultiSOM method include reducing the noise which is inevitably generated in an overall classification approach while increasing the flexibility and granularity of the analyses.

With this potential usefulness, we started to try from the basic SOM by use of the tool developed by Kleiweg (Extended Kohonen maps). We used the default setting of the tool and varied the map size from 8 × 8, 10 × 10, 16 × 16, to 34 × 34. Unfortunately, all these maps could not lead us to a meaningful result for this NSC patent set. The resulting SOMs do not show desired clustering when each patent was labeled with the academic divisions introduced above. Most divisions scatter all over the map. Even the most unique Biotechnology has outliers in different locations. In the end, no further variation of SOM was tried.

In another attempt to compare our results with others', we also have tried to cluster these NSC patents based on their co-cited and co-citing intensity. But only 123 (among 612) patents are co-cited by others resulting in 99 co-cited pairs and only 175 (among 612) patents co-cite others resulting in 143 co-citing pairs. Such sparseness may lead to biased analysis. Therefore, citation analysis is not suitable in this case.

In short, we have tried to compare our approach with others, but this effort has not yet led to meaningful results.

7. Conclusions and future work

This paper describes a series of mining techniques that conforms to the analytical process adopted by and used to train patent analysts (at least in Taiwan, see Chen, 1999). The automation of the whole process not only helps create final patent maps for topic analysis, but also facilitates other patent analysis tasks because each step in this process has its own application. For example, after segmentation, more effective classification can be achieved by combining individual segment matching rather than by whole patent matching (Kim, Huang, Jung, & Choi, 2005). After abstraction, the topics, functions, or technologies revealed in each patent can be more easily accessed and shared by the analysts. After keyword extraction and co-word analysis, relevant terms can be suggested to users in prior art search to relieve the human burden in devising domain-specific search terms. After clustering, the patent collection is organized in a way that may complement the IPC or UPC partition of the collection for analytic purpose. And finally, visualization is a way to combine all these results to suggest patterns, relations, or trends.

In the design of these techniques, we have proposed a rigorous approach to verify the usefulness of segment extracts as the document surrogates, a dictionary-free keyphrase extraction algorithm, an efficient co-word analysis method, and an automatic procedure to produce generic cluster titles. Evaluation of these techniques has shown some success.

Nevertheless, more patent-related text mining issues can be studies. Unlike scientific publications, the regular style of the patent documents deserves more attention. So far we only use the summaries of the segments for document surrogates. The functions of each segment have yet to be explored. For example, one may use only the background segment for domain analysis, since it covers the domain background of a patent, not the details of it. Moreover, one may extract the problem to be solved from the background segment and major solution methods from the summary of the invention with more sophisticated summarization techniques. By clustering the problems and solutions individually like the patent mapping task in the NTCIR Workshop 4, an analytic map similar to the technology-effect matrix can be created for a domain of patents. Our future work may explore this direction in using the patent segments.

Acknowledgements

This work is supported in part by NSC under the grants numbered: NSC 93-2213-E-030-007- and NSC 94-2524-S-003-014-.

Appendix A. The clue words for patent summarization are listed below. These words are mainly for the background segment.

Advantage
Avoid
Cost
Costly
Decrease
Difficult
Effectiveness
Efficiency
Goal
Important
Improved
Increase
Issue
Limit
Needed
Overhead
Performance
Problem
Reduced
Resolve
Shorten
Simplify
Suffer
Superior
Weakness

Appendix B

The background segment of a US patent and its summaries. Those best sentences selected by Microsoft Word are in italic font, while those selected by our method is in boldface.

TITLE (Patent No.: 6,519,591)

Vertical implementation of expectation-maximization algorithm in SQL for performing clustering in very large databases.

BACKGROUND OF THE INVENTION

Relational databases are the predominate form of database management systems used in computer systems. *Relational database management systems are often used in so-called "data warehouse" applications where enormous amounts of data are stored and processed.* In recent years, several trends have converged to create a new class of data warehousing applications known as data mining applications. Data mining is the process of identifying and interpreting patterns in databases, and can be generalized into three stages.

Stage one is the reporting stage, which analyzes the data to determine what happened. Generally, most data warehouse implementations start with a focused application in a specific functional area of the business. These applications usually focus on reporting historical snap shots of business information that was previously difficult or impossible to access. Examples include Sales Revenue Reporting, Production Reporting and Inventory Reporting to name a few.

Stage two is the analyzing stage, which analyzes the data to determine why it happened. As stage one end-users gain previously unseen views of their business, they quickly seek to understand why certain events occurred ; for example a decline in sales revenue. After discovering a reported decline in sales, data warehouse users will then obviously ask, "Why did sales go down?" Learning the answer to this question typically involves probing the database through an iterative series of ad hoc or multidimensional queries until the root cause of the condition is discovered. *Examples include Sales Analysis, Inventory Analysis or Production Analysis.*

Stage three is the predicting stage, which tries to determine what will happen. As stage two users become more sophisticated, they begin to extend their analysis to include prediction of unknown events. For example, "Which end-users are likely to buy a particular product", or "Who is at risk of leaving for the competition?" It is difficult for humans to see or interpret subtle relationships in data, hence as data warehouse users evolve to sophisticated predictive analysis they soon reach the limits of traditional query and reporting tools. Data mining helps end-users break through these limitations by leveraging intelligent software tools to shift some of the analysis burden from the human to the machine, enabling the discovery of relationships that were previously unknown.

Many data mining technologies are available, from single algorithm solutions to complete tool suites. Most of these technologies, however, are used in a desktop environment where little data is captured and maintained. Therefore, most data mining tools are used to analyze small data samples, which were gathered from various sources into proprietary data structures or flat files. On the other hand, organizations are beginning to amass very large databases and end-users are asking more complex questions requiring access to these large databases.

> Unfortunately, most data mining technologies cannot be used with large volumes of data. Further, most analytical techniques used in data mining are algorithmic-based rather than data-driven, and as such, there are currently little synergy between data mining and data warehouses. Moreover, from a usability perspective, traditional data mining techniques are too complex for use by database administrators and application programmers, and are too difficult to change for a different industry or a different customer.
>
> One analytic algorithm that performs the task of modeling multidimensional data is "cluster analysis". Cluster analysis finds groupings in the data, and identifies homogenous ones of the groupings as clusters. If the database is large, then the cluster analysis must be scalable, so that it can be completed within a practical time limit.
>
> **In the prior art, cluster analysis typically does not work well with large databases due to memory limitations and the execution times required**. **Often, the solution to finding clusters from massive amounts of detailed data has been addressed by data reduction or sampling, because of the inability to handle large volumes of data**. However, data reduction or sampling results in the potential loss of information.
>
> **Thus, there is a need in the art for data mining applications that directly operate against data warehouses, and that allow non-statisticians to benefit from advanced mathematical techniques available in a relational environment**.

References

ACL-2003 Workshop on Patent Corpus Processing, 12 July 2003, Sapporo, Japan. http://www.slis.tsukuba.ac.jp/~fujii/acl2003ws.html.

ACM SIGIR 2000 Workshop on Patent Retrieval. http://research.nii.ac.jp/ntcir/sigir2000ws/.

Archibugi, D., & Pianta, M. (1996). Measuring technological change through patents and innovation survey. *Technovation, 16*(9), 451–468.

Bay, Y.-M. (2003). Development and applications of patent map in Korean high-tech industry. *The first Asia-Pacific conference on patent maps, Taipei, October 29*, pp. 3–23.

Be'de'carrax, C., & Huot, C. (1994). A new methodology for systematic exploitation of technology databases. *Information Processing & Management, 30*(3), 407–418.

Bekkerman, R., El-Yaniv, R., Winter, Y., & Tishby, N. (2001). On feature distributional clustering for text categorization. In *Proceedings of the 24th annual international ACM-SIGIR conference on research and development in information retrieval*, pp. 146–153.

Booker, A., Condliff, M., Greaves, M., holt, F. B., Kao, A., Pierce, D. J., et al. (1999). Visualizing text data sets. *IEEE Computing in Science and Engineering, 1*(4), 26–34.

Brown, P. F., Della Pietra, V. J., De Souza, P. V., Mercer, R. L., & Lai, J. C. (1992). Class-based N-gram models of natural language. *Computational Linguistics, 18*(4), 467–479.

Campbell, R. S. (1983). Patent trends as a technological forecasting tool. *World Patent Information, 5*(3), 137–143.

Chen, B. (1999). Introduction to patent map. *Lecture notes for the training of patent mapping and patent analysis*. Taipei, National Science Council (in Chinese).

Choueka, Y. (1988). Looking for needles in a haystack, or locating interesting collocational expressions in large textual databases. In *Proceedings of the RIAO*, pp. 609–623.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41*, 391–407.

Document Understanding Conferences. http://www-nlpir.nist.gov/projects/duc/.

Ernst, H. (1997). Use of patent data for technological forecasting: the diffusion of CNC-technology in the machine tool industry. *Small Business Economics, 9*, 361–381.

Fagan, J. L. (1989). The effectiveness of a nonsyntactic approach to automatic phrase indexing for document retrieval. *Journal of the American Society for Information Science, 40*(2), 115–132.

Fall, C. J., Torcsvari, A., Benzineb, K., & Karetka, G. (2003). Automated categorization in the international patent classification. *ACM SIGIR Forum, 37*(1), 10–25.

Fang, H., Tao, T., & Zhai, C. X. (2004). A formal study of information retrieval heuristics. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval, Sheffield, United Kingdom*, pp. 49–56.

Fattori, M., Pedrazzi, G., & Turra, R. (2003). Text mining applied to patent mapping: a practical business case. *World Patent Information, 25*, 335–342.

Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., & Uthurasamy, R. (1996). *Advances in knowledge discovery and data mining*. AAAI Press/The MIT Press.

Frakes, W. B., & Baeza-Yates, R. (1992). *Information retrieval: Data structure and algorithm*. Prentice Hall.

Frank, E., Hall, M., & Trigg, L. Weka 3: Data Mining Software in Java. http://www.cs.waikato.ac.nz/ml/weka/.

Fujii, A., Iwayama, M., & Kando, N. (2004). Overview of patent retrieval task at NTCIR-4. In *Proceedings of the fourth NTCIR workshop on evaluation of information retrieval, automatic text summarization and question answering, June 2–4, Tokyo, Japan*.

Glenisson, P., Glanzel, W., Janssens, F., & De Moor, B. (2005). Combining full text and bibliometric information in mapping scientific disciplines. *Information Processing & Management, 41*(6), 1548–1572.

Hearst, M.A. (1999). Untangling text data mining. In *Proceedings the 37th annual meeting of the association for computational linguistics, June 20–26*, pp. 3–10.

Information Mapping Project. Computational Semantics Laboratory, Standford University. http://infomap.stanford.edu/.

Iwayama, M., Fujii, A., Kando, N., & Marukawa, Y. (2006). Evaluating patent retrieval in the third NTCIR workshop. *Information Processing and Management, 42*, 207–221.

Jain, A. K., Murthy, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Reviews, 31*(3), 264–323.

Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the European conference on machine learning, Berlin*, pp. 137–142.

Jung, S. (2003). Importance of using patent information. In *WIPO—Most intermediate training course on practical intellectual property issues in business, organized by the World Intellectual Property Organization (WIPO), Geneva, November 10–14*.

Karypis, G. CLUTO: a software package for clustering low- and high-dimensional datasets. http://www-users.cs.umn.edu/~karypis/cluto/.

Kim, J.-H., Huang, J.-X., Jung, H.-Y., & Choi, K.-S. (2005). Patent document retrieval and classification at KAIST. In *Proceedings of the fifth NTCIR workshop on evaluation of information access technologies: information retrieval, question answering, and cross-lingual information access, Tokyo, Japan, December 6–9*, pp. 304–311.

Kleiweg, P. Extended Kohonen maps. http://www.let.rug.nl/~kleiweg/kohonen/kohonen.html.

Kleiweg, P. Software for dialectometrics and cartography. http://www.let.rug.nl/~kleiweg/L04/.

Kohonen, T. (1997). *Self-organizing maps*. Secaucus, NJ: Springer-Verlag New York, Inc..

Kruskal, J. B. (1977). Multidimensional scaling and other methods for discovering structure. In K. Enslein, A. Ralston, & H. S. Wilf (Eds.), *Statistical methods for digital computers* (pp. 296–339). New York: Wiley.

Kupiec, J., Pedersen, J. O., & Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th international ACM SIGIR conference on research and development in information retrieval*, pp. 68–73.

Lagus, K., Kaski, S., & Kohonen, T. (2004). Mining massive document collections by the WEBSOM method. *Information Sciences, 163*(1–3), 135–156.

Lai, K.-K., & Wu, S.-J. (2005). Using the patent co-citation approach to establish a new patent classification system. *Information Processing & Management, 41*(2), 313–330.

Lamirel, J.-C., Shadi Al Shehabi, Hoffmann, M., & Francois, C. (2003). Intelligent patent analysis through the use of a neural network: experiment of multi-viewpoint analysis with the MultiSOM model. In *Proceedings of the ACL workshop on patent corpus processing, Sapporo, Japan, 12 July*.

Larkey, L. S. (1999). A patent search and classification system. In *Proceedings the 4th ACM conference on digital libraries*, pp. 179–187.

Lent, B., Agrawal, R., & Srikant, R. (1997). Discovering trends in text databases. In *Proceedings of international conference on knowledge discovery and data mining, Newport Beach, California, USA, August 14–17*.

Liu, S.-J. (2003). Patent map – a route to a strategic intelligence of industrial competitiveness. In *The first Asia-Pacific conference on patent maps, October 29, Taipei*, pp. 2–13.

Losiewicz, P., Oard, D. W., & Kostoff, R. N. (2000). Textual data mining to support science and technology management. *Journal of Intelligent Information Systems, 15*(2), 99–119.

Mai, F.-D., Hwang, F., Chien, K.-m., Wang, Y.-M., & Chen, C.-y. (2002). Patent map and analysis of carbon nanotube. Science and Technology Information Center, National Science Council, ROC.

Mani, I. (2001). *Automatic summarization*. John Benjamins.

Ng, H. T., Goh, W. B., & Low, K. L. (1997). Feature selection, perception learning, and a usability case study for text categorization. In *Proceedings of the 20th international ACM-SIGIR conference on research and development in information retrieval*, pp. 67–73.

Noyons, E. C. M., & Buter, R. K. CWTS Bibliometric Mapping Projects. http://www.cwts.nl/ed/projects/home.html.

Noyons, E. C. M., & van Raan, A. F. J. (1998a). Mapping scientometrics, informetrics, and bibliometrics. CWTS Working papers. http://www.cwts.nl/ed/sib/.

Noyons, E. C. M., & van Raan, A. F. J. (1998b). Advanced mapping of science and technology. *Scientometrics, 41*, 61–67.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*(3), 130–137.

Robertson, S. E., & Walker, S. (1994). Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on research and development in information retrieval, Dublin, Ireland, July 03–06, pp. 232–241*.

Salton, G. (1989). *Automatic text processing: The transformation, analysis, and retrieval of information by computer*. MA: Addison-Wesley.

Schutze, H. (1993). Word space. In S. J. Hanson, J. D. Cowan, & C. L. Giles (Eds.). *Advances in neural information processing systems* (vol. 5, pp. 895–902). Morgan Kaufman.

Sheremetyeva, S. (2003). Natural language analysis of patent claims. *ACL workshop on patent corpus processing, Sapporo, Japan, 12 July*.

Shinmori, A., Okumura, M., Marukawa, Y., & Iwayama, M. (2003). Patent claim processing for readability - structure analysis and term explanation. In *ACL workshop on patent corpus processing, Sapporo, Japan, 12 July*.

Singhal, A., Buckley, C., & Mitra, M. (1996). Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval, Zurich, Switzerland*, pp. 21–29.

Smadja, F. (1993). Retrieving collocations from text: Xtract. *Computational Linguistics, 19*, 143–177.

The 8th Science and Technology Foresight Survey – Study on Rapidly-Developing Research Areas – Interim Report, Science and Technology Foresight Center, National Institute of Science & Technology Policy, Japan, 2004. http://www.nistep.go.jp/index-e.html.

Tseng, Y.-H. (1999). Content-based retrieval for music collections. In *Proceedings of the 22nd international ACM SIGIR conference on research and development in information retrieval, August 15–19, Berkeley, USA*, pp. 176–182.

Tseng, Y.-H. (2002). Automatic thesaurus generation for Chinese documents. *Journal of the American Society for Information Science and Technology, 53*(13), 1130–1138.

Tseng, Y.-H., Juang, D.-W. (2003). Document-self expansion for text categorization. In *Proceedings of the 26th international ACM SIGIR conference on research and development in information retrieval*, pp. 399–400.

Tseng, Y.-H., Juang, D.-W., & Chen, S.-H. (2004). Global and local term expansion for text retrieval. In *Proceedings of the fourth NTCIR workshop on evaluation of information retrieval, automatic text summarization and question answering, June 2–4, Tokyo, Japan*.

Tseng, Y.-H., Juang, D.-W., Wang, Y.-M., & Lin, C.-J. (2005). Text mining for patent map analysis. In *Proceedings of IACIS Pacific 2005 conference, May 19–21, Taipei, Taiwan*, pp. 1109–1116.

Uchida, H., Mano, A., & Yukawa, T. (2004). Patent map generation using concept-based vector space model. In *Proceedings of the fourth NTCIR workshop on evaluation of information access technologies: information retrieval, question answering, and summarization, June 2–4, Tokyo, Japan*.

United States Patent and Trademark Office. http://www.uspto.gov/.

van Rijsbergen, K. Information retrieval. http://www.dcs.gla.ac.uk/Keith/Chapter.2/Table_2.1.html.

WordNet: a lexical database for the English language, Cognitive Science Laboratory Princeton University. http://wordnet.princeton.edu/.

Yang, Y., Ault, T., Pierce, T., & Lattimer, C. W. (2000). Improving text categorization methods for event tracking. In *Proceedings of the 23rd annual international ACM-SIGIR conference on research and development in information retrieval*, pp. 65–72.

Yang, Y., Liu, X. (1999). A re-examination of text categorization methods. In *Proceedings of the 22nd annual international ACM-SIGIR conference on research and development in information retrieval*, pp. 42–49.

Yang, Y., Pedersen, J. (1997). A comparative study on feature selection in text categorization. In *Proceedings of the international conference on machine learning*, pp. 412–420.

Ye, J.-Y. (2004). Evaluation of term suggestion in an interactive Chinese retrieval system. Master Thesis, Department of Library and Information Science, Fu Jen Catholic University.

Yoon, B., & Park, Y. (2004). A text-mining-based patent network: analytical tool for high-technology trend. *Journal of High Technology Management Research, 15*, 37–50.