

Using the Web to Construct Taxonomy for a Heterogeneous Community of Practice

Isak Taksa

*Zicklin School of Business
Baruch College
City University of New York
One Bernard Baruch Way
New York, NY 10010, USA*

Amanda Spink

*Faculty of Information Technology
Queensland University of Technology
Gardens Point Campus
2 George St, GPO Box 2434
Brisbane QLD 4001 Australia*

Abstract

A heterogeneous community of practice spans many disciplines, industries and professions. Members of these communities are united by common research, products and experiences but are frequently separated by specialized vocabulary and industry terms. This lack of language commonality presents a challenge to efficiently locating relevant Web based information which usually depends on the user's knowledge of the field and ability to select suitable terms to formulate a search query. Research and practice have shown that the quality of information retrieval is significantly improved when taxonomy is employed to organize terms that describe the search domain. This paper presents an innovative, collaborative approach to building taxonomy for a particular domain, populating it with Web content and sharing it among members of the community of practice. A model is built and results and implications are discussed.

Keywords: information retrieval, taxonomy, text categorization, digital library

1. Introduction

Communities of practice (CoP) are established by experts who share a common profession, similar practices and universal language. This harmony facilitates efficient information acquisition, sharing and management. However, as more and more industries and professions fuse knowledge and human resources to accomplish more complex goals, CoP expand with the influx of new members bringing new skills and new specialty languages. Information acquisition becomes compartmentalized and sharing fades away.

These issues are intensely manifested in a rapidly expanding patent processing CoP. Scientific and technological progress has created a need for patents. Innovators, practitioners, and scientists around the world

responded with over a million filings inundating national patent offices last year alone [11]. The information required by these offices to prepare new claims and subsequently create new patents is enormous in variety of forms and specialized vocabularies. While essential information is largely available on the Web, finding it frequently becomes a difficult task caused by several issues: cross-language issues in patent retrieval, lack of information organization (taxonomy) and structure for collaboration. Reuse of retrieved information or exchange of acquired knowledge by the patent CoP is also virtually nonexistent.

In this paper we demonstrate an approach to collecting Web resources and constructing taxonomy for a patent processing community. We discuss the advantages of using natural language in conjunction with our approach to long query collaborative information retrieval (CIR), and the retention of search results. We explain a process of using a matrix space model to populate taxonomy of the patent processing domain and conclude with a discussion and future research directions.

2. Collaborative information retrieval (CIR)

Information Retrieval (IR) is usually considered an individual endeavor. This is especially true for professionals, who are expert in their respective areas and know how to search and find relevant information and therefore don't seek help or collaboration. When it comes to the patent domain, which spans many disciplines and industries, collaboration in IR is a required and valuable skill [5]. CIR is a mature field with many researchers stressing different aspects of collaboration. There are several social issues of sharing information: for example, some information seekers consistently share search retrieval results with others in the confinement of a well defined group, but withhold individual information, such as actual queries used to

obtain information or relevancy judgment of these results, or at the other extreme (typical for scholars), information seekers who arbitrarily share various amounts and types of results depending on the composition and purpose of a loosely defined group. In our case, as demonstrated later, the engine is designed for CIR and all queries and respective results are available for review and analysis by all participants.

2.1. Accepting and processing users' needs

A way to insure proper conversion of user problem oriented information needs (POIN) into an adequate query is to allow the user to input a query expressed in a natural language without any limitation on size or form of the expression. But many search engines do not expect (and certainly do not encourage) the user to enter more than a few terms, regardless of a user's information need. Bearing in mind that the length of an average search query still lingers between 2 and 3 terms [6, 9], it is usual that a user has to look for the right terms to iteratively formulate many various queries while refining the original search query for the most relevant results.

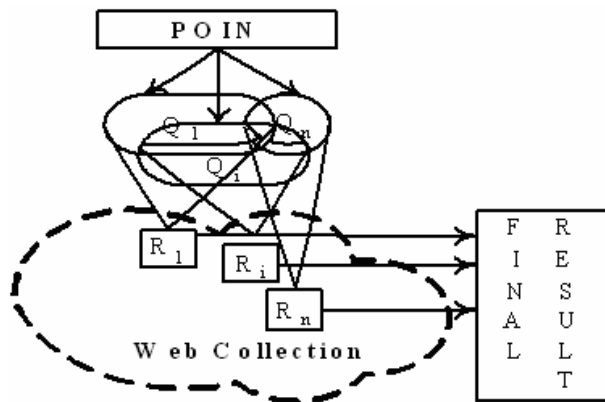


Figure 1. Metasearch engine data flow

Moreover, most users are not sufficiently trained to select the best terms to get the results that they seek. To alleviate this problem the metasearch engine we designed accepts an original verbose description of a user's information need and formulates multiple smaller queries acceptable by most search engines.

The idea behind multiple query representation is quite simple – create intersecting instances (subqueries SQ_i) of the original description, use these subqueries to conduct individual searches and collect interim results (R_i), and then merge all interim results creating one ranked list (Figure 1 above illustrates this

process). Unlike other query expansion techniques common to IR that require additional language resources, our approach is based solely on the original user query.

To demonstrate this concept we use an abstract extracted from a scientific paper. To begin the search process we “cut” the abstract of the above paper and “paste” it (entire abstract) into the search field of our search engine. Next, the abstract is broken into individual terms; “stop” words and duplicate words are removed. After search terms are determined we calculate each term's weight using $tf*idf$ measure (tf – term frequency in the given document, idf – inverse document frequency in the entire collection). There are many variations of tf and idf measures and their combinations to obtain the term's weight. However an earlier research [12] has shown that the choice of different variations of $tf*idf$ measures and/or their combinations have relatively little effect on the quality of search results when averaged over many queries. Since all our IR experiments use Google as the underlying search engine, we associated the weight of each term with its Google document frequency. The list of search terms, sorted in reverse order of their frequencies, is presented to the user via a user interface as a suggested list of search terms for review and (potential) modification.

User interface and feedback in query term selection are important functions. This is especially true for CIR. While some researchers stress group participation in the query reformulation process others emphasize continuous interactive query refinement. User control is especially important over totally automated functions. Since the query parsing and terms ordering is performed without the user's participation we provide this interface to leave to the user the final decision regarding importance and rank of search terms. This is a continuous process and could be performed repeatedly. These search terms are then used to formulate multiple sub-queries (see Figure 2 below).

2.2. Formulating queries and merging results

The metasearch engine creates multiple intersecting subqueries from top terms in the ordered list of terms. The process we use was introduced in earlier research [8] and is quite straightforward - create various nCr combinations - or simply, create all possible conjunctive search queries consisting of at least r terms from the list of n terms. The original research suggests that the best results are achieved for the following values of n and r : $7 \leq n \leq 9$ and $3 \leq r \leq 6$. For our experiments we selected $n = 9$ and $r = 5$. Later research [10] demonstrated that depending on the mix of terms some of the above combinations could be skipped to expedite the process without any performance degradation.

Earlier studies [1] investigated the effect of progressive combination of multiple representations of TREC topics on IR performance and they also compared

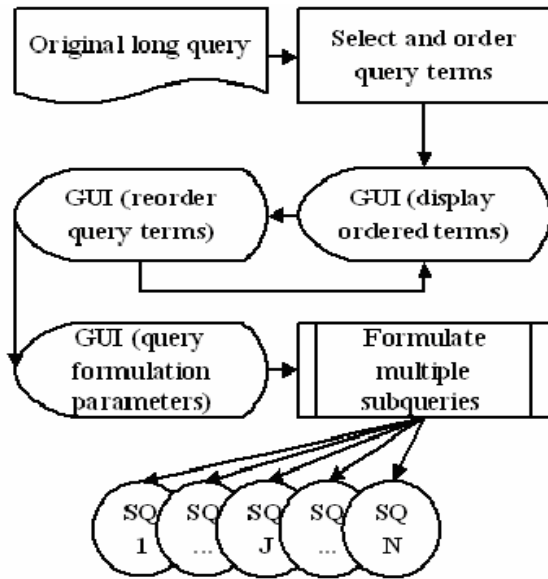


Figure 2. Subquery formulation process

two different approaches to data fusion: score vs. rank. When dealing with query combination, the rules used were based on similarity scores between a topic and a document. On the other hand, when dealing with multiple results from different systems, combinations were based on rank information. While early studies favored the similarity method, the later ones favored the ranking method [7]. For this experiment we used a modified version of Single Engine Fusion (SEF) rank algorithm [8]. The primary consideration to determine the final position of an URL in the final ranked list was how many interim lists contain this URL.

2.3. Analyzing and saving final results

We limit the final list to 30 results (80 % of users never look beyond the first page as reported earlier [9]). The user then opens each document and decides its relevancy. After reviewing all documents the user saves the relevant results. At this or any later time the user can access the file and extract the abstract. In case of a non-scientific article (no abstract available) the first 1000 characters are saved in the data base. Search terms used to retrieve relevant results are also stored for subsequent use in building and populating the taxonomy.

3. Building the taxonomy

In the prior section we described the process of collecting assorted abstracts from the Internet domain of

patent processing. To make the content of this collection easy to use and valuable to the patent community of practice we created taxonomy of the patent processing domain and categorized collected abstracts for easy browsing and retrieval. This taxonomy will allow continuous reuse of the content, saving time and effort required to retrieve the essential information time and again. Designing and applying the patent domain taxonomy is a three-step process (see Figure 3 below):

1. selecting terms for the controlled dictionary;
2. building taxonomy;
3. categorizing the content.

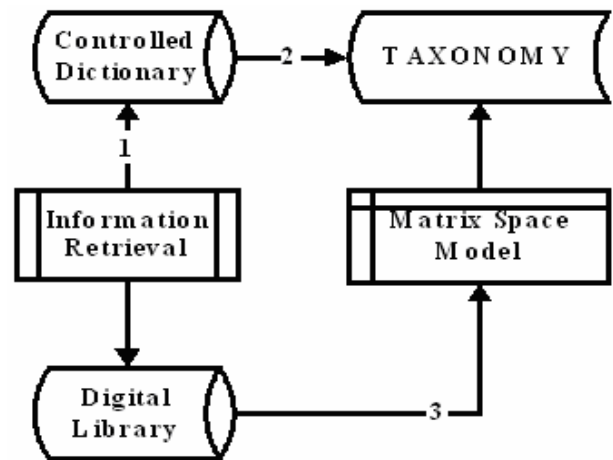


Figure 3. Building and populating taxonomy

Selecting concepts and terms for the controlled vocabulary involves examining the content of popularly used electronic and printed resources. What makes this task unusually challenging for the patent domain is the diversity of stakeholder groups and their goals. The language used by these groups reflects this diversity. When innovators talk about patent filing and royalties, their attorneys hear patent prosecution and management. When searching numerous national and international databases to formulate *prior art* preparers worry about patents in foreign languages, while the scientists respond with research in cross-language or multilingual patent retrieval.

Our first approach was to determine whether there is an existing taxonomy of the patent domain with its controlled vocabulary. With the exception of the IPC (International Patent Classification) that covers close to 70,000 categories of patents (not patent processing), we found nothing. For example, the North American Industry Classification System has only 6 distinct codes for patent related businesses in its index entries, and none in its Codes and Titles section. Finding no existing taxonomy, for this research we analyzed over 300 academic and industrial publications from the patent

domain using title, abstracts, or the first 1000 characters (for non academic documents). We selected a small sample of 64 patent domain terms for use in our experiment.

3.1. Structuring the taxonomy

Our next step required a fundamental decision – which taxonomy model to adapt: descriptive (content-based) or navigational (knowledge-based) [2]. While the former utilizes precise definitions of terms and assists users in searching and retrieving new information, the second allows for more ambiguous definitions of terms since users have an idea of what they are looking for, and supports discovery of new information through browsing of an existing collection. The advantage of a descriptive taxonomy, when properly structured and sufficiently populated, is that it can span many diverse domains allowing for a common vocabulary to the entire community of practice. Populating the taxonomy requires reviewing all potential documents against a controlled vocabulary, looking for similarities and relationships.

We built our taxonomy as a browsing version of a descriptive taxonomy. This hybrid taxonomy is content-based and clearly understood by all potential users. It is descriptive of the content and displays a visual path to it. Each path ends in a leaf that links to a file with the appropriate content. We represent the taxonomy as a machine-readable table (see Table 1 below). This table is used by the taxonomy generation program to generate the visualization of the taxonomy.

The file name in column 3 (see Table 1) reflects the organization of the table and the position of the term in the taxonomy.

For example, the term *Structures* (Item # 11 on the list) is associated with a file **1010105030** (five two-digit numbers) which means that this term is a fifth level term under the terms *Methods* (#8), *IR* (#3), *Research* (#2), *Patent Processing* (#1). This file naming convention allows for easier insertion of new terms when expanding the taxonomy and for associating a vector for each term in the taxonomy (next section). The user, via the interface, can visualize a path to any term defined in the taxonomy and by clicking the associated file name will gain access to the top level content.

To categorize our collection according to the newly based taxonomy we utilize a matrix space model (MSM), originally used for classification of military documents [4]. In this model the authors described both documents and the queries as a matrix of concepts that were later analyzed for relevancy. Subsequently, another group of researchers [3] applied MSM to classic IR (computing similarity between a query described as a matrix and a document). Our approach to MSM is different.

Table 1. Building and populating taxonomy

Id	Term	File Name
1	Patent Processing	10
2	Research	1010
3	Information Retrieval	101010
4	Rankings	10101010
5	Relevance Feedback	10101020
6	Monolingual	10101030
7	Cross Lingual	10101040
8	Methods	10101050
9	Terms/Sentences	1010105010
10	Queries	1010105020
11	Structures	1010105030
12	Tools/Models	101030
13	R2D2	10103010
14	MEISTER	10103020
15	Topics	101040
16	Text Summarization	10104010
17	Visualization	10104020
18	Evaluation	10104030
19	Case Study	10104040
20	Swedish PO	1010404010
21	Experiments	101050
...
63	Reexamination/Interference	10902030
64	Management	109030

We use a matrix to describe our entire taxonomy, a vector to describe our document and their product to determine the top potential category (categories) to classify each document. We will now describe the process in detail.

3.2. Taxonomy as a matrix

Every term in the taxonomy (and its path to the root via other terms) is encoded by a row-vector in a binary matrix $T \in \{0,1\}^{N \times N}$ (N is the number of terms in the taxonomy, here 64), which describes this term. Each position corresponds to the position of the item number associated with this term. Then, the unique path to the root is traversed by following the number pairs in the file name of the original term, and for each node visited, the corresponding bit in the vector is “turned on”. For example (refer to Table 1), *Swedish PO* (#20) has file name **1010404010** associated with that term on level 5. A vector is created and initially bit 20 is set on. Its parent on level four will have file name **10104040** associated with it and corresponds to *Case Study* #19); therefore, bit 19 is set on. Continuing in this manner until the root of the taxonomy tree, terms with

associated file names **101040** (*Topics*, #15), **1010** (*Research*, #2), and **10** (*Patent Processing*, #1) will be visited and the corresponding bit positions 15, 2 and 1 will be set on. This process is repeated for all terms in Table 1 and results in the matrix T (see Figure 4).

3.3 Document as a vector

Encoding the document as a vector is more complex. While the taxonomy utilizes a controlled vocabulary (albeit limited in size), the documents are described in natural language and do not easily “fit” into a vector. To represent documents in the collection as vectors, a semantic reduction is performed (this process translates documents in natural language into the limited controlled vocabulary). Our approach, while labor intensive, was to develop a cross-referenced table for keyword/phrase substitution. For example, text in the document collection referring to multiple languages (bilingual, multilingual, etc.) is replaced by a *Cross-Lingual* (#7). While this approach is imperfect due to the richness of natural language, further expansion of the controlled vocabulary and application training combined with user feedback will significantly improve this approach. Once the reduction process is completed, the document vector D is created, consistent with the process of constructing the taxonomy matrix.

3.4. Determining similarity

In classical IR, similarity between documents is calculated as a cosine between two vectors representing a query and a document. This research calculates similarity between a document and the taxonomy by multiplying the taxonomy matrix $T \in \{0,1\}^{N \times N}$ by the document vector $D \in \{0,1\}^N$, resulting in a similarity vector $S \in \{0, \dots, k\}^N$ (see Figure 4 below).

Each component of the resulting vector (s_i) represents similarity (number of concept co-occurrence) between the document and the i^{th} branch of the taxonomy. For example, s_{20} would represent similarity between a given document and the branch of the taxonomy from *Swedish PO* back to the root (*Patent processing*). By selecting the largest s_i ($\max(s_i)$) we determine category i that the document best belongs to, and then insert the document reference into the file associated with that category (in the case of a tie the term at the lowest level wins). In the case of $i=20$ (*Swedish PO*) the document would be added to file **1010404010** (see Table 1).

Since this similarity method considers the importance of the concept to the taxonomy more than the document, we compensate by assigning documents to the top three categories.

$$T * D = \begin{bmatrix} t_{1,1} & t_{1,2} & \cdots & t_{1,j} & \cdots & t_{1,N} \\ t_{2,1} & t_{2,2} & \cdots & t_{2,j} & \cdots & t_{2,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{i,1} & t_{i,2} & \cdots & t_{i,j} & \cdots & t_{i,N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ t_{N,1} & t_{N,2} & \cdots & t_{N,j} & \cdots & t_{N,N} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_i \\ \vdots \\ d_N \end{bmatrix} = S = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_i \\ \vdots \\ s_N \end{bmatrix}$$

T D S

Figure 4. Taxonomy matrix

To analyze effectiveness of such overcompensation, the rank of categorization (first, second, third) is stored in the document record. When a user views a file associated with a category, and decides that a particular document does not belong to this category, the user can mark the document for deletion. At a later time, the taxonomy administrator will analyze all documents marked for deletion, and decide how to modify the categorization process.

4. Discussion

This paper proposes and investigates a concept of knowledge acquisition and sharing in the patent community of practice (CoP). The model, developed to prove this concept, demonstrates how individuals and groups collaborate and share results.

4.1. Using natural language for CIR

The collaborative search engine designed for retrieval experiments allows natural language to express the user’s information needs and accepts a search query of any length. This is especially important for inexperienced users who are at a loss for search terms, or for users conducting a search in an unfamiliar domain. Users benefit from the collaborative functionality of the search engine. Relevant results, and original search queries used to obtain them, are stored at the server level. Search terms used to retrieve relevant documents are available to all. Documents are retrieved, evaluated for relevancy and downloaded once, thus saving time on repeating search and retrieval tasks for the same documents, a common drawback when many individuals conduct similar searches.

4.2. Populating and employing taxonomy

We use a matrix space model combined with a controlled vocabulary to categorize retrieved content

and to populate the partial taxonomy of the patent domain built for the experiments. This taxonomy represents a partial map of the small sample of Web patent resources. A flexible user interface offers easy navigation for novice and expert alike to any category of knowledge on this map. Once the user picks a category, the content of this category is presented via a list of titles, abstracts or full-length documents. This systematic, multistage access to relevant content improves the efficiency of navigation.

4.3. Building Community of Practice

An important benefit of our approach is the participation of the community – a heterogeneous community known for possessing tacit knowledge not easily shared among its members. Experts in a particular field know how to locate and evaluate field specific knowledge on the Web. Using controlled vocabulary developed by the community, this knowledge is classified and subsequently shared by all members triggering discovery of additional knowledge.

5. Conclusion and further research

Our research demonstrated how patent processing taxonomy built from Web resources could be used by members of the patent CoP to jointly share and create knowledge. Further research would expand on topics addressed in this paper.

Currently the query formulation parameters are set by users based on their individual experience. One possible route for improvements is utilizing frequencies of search terms in the search engine collection to dynamically set these parameters. Recent research supports this suggestion.

The other, potentially beneficial area for improvements is the quality of search results. At present, users have manual access to search results and search terms used by others. We intend to analyze search results found to be relevant by a group of users and build a recommender system to assist with search term selection and relevance judgment. Another potential direction for research is the use of the expanding taxonomy for user supervised query expansion.

As the number of documents in the taxonomy continues to grow and more documents are categorized, there is a need for topic detection/splitting algorithm. This algorithm will allow taxonomy restructuring (tree growing) without requiring the entire document collection to be categorized from scratch.

6. References

- [1] N.J. Belkin, P.B. Kantor, C. Cool and R. Quatrain, "Combining evidence for information retrieval", In D. Harman (Ed.), *TREC-2, Proceedings of the Second Text Retrieval Conference*, GPO, Washington, D.C., 1994, pp. 35 – 44.
- [2] S. Conway and C. Sigar, *Unlocking Knowledge Assets*, Redmond, Microsoft Press, 2002.
- [3] K. Gao, Y. Wang, and A.Z. Wang, "An efficient relevant evaluation model in information retrieval and its application", *The Fourth International Conference on Computer and Information Technology (CIT'04)*, 2004, pp. 845 – 850.
- [4] A. Guitouni, A-C. Boury-Brisset, L. Belfares, K. Tiliki, N. Belacel, C. Pourier and P. Biloudaeu, "Automatic documents analyzer and classifier", *Proceedings of the 7th International Command and Control Research and Technology Symposium*, Québec City, QC, CA, 2002, NRC Publication # 44987.
- [5] P. Hansen and K. Järvelin, "Collaborative information retrieval in an information-intensive domain", *Information Processing & Management*, 2005, Vol.41(5), pp. 1101 – 1119.
- [6] B.J. Jansen and A. Spink, "How are we searching the World Wide Web? A comparison of nine search engine transaction logs", *Information Processing & Management*, Vol. 42/1, 2006, pp. 248-263.
- [7] J.H. Lee, "Analysis of multiple evidence combination", *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, PA, 1997, pp. 267 – 275.
- [8] J. Shapiro and I. Taksa, "Constructing web search queries from the user's information need expressed in a natural language", *Proceedings of the 2003 ACM Symposium on Applied Computing*, Melbourne, FL, 2003, pp. 1157 – 1162.
- [9] A. Spink, and B.J. Jansen, *Web Search: Public Searching of the Web*, Berlin, Springer, 2004.
- [10] I. Taksa, "Predicting the cumulative effect of multiple query formulations", *International Symposium on Information Technology: Coding and Computing (ITCC 05)*, Las Vegas, NV, 2005, Vol. 2, pp. 491 – 496.
- [11] WIPO, Press Release 401, Geneva, January 14, 2005, http://www.wipo.int/edocs/prdocs/en/05/wipo_pr_05_401.html [Last accessed December 15, 05]
- [12] J. Zobel and A. Moffat, "Exploring the similarity space", *ACM SIGIR Forum*, 1998 Vol. 35(1), pp. 18 – 34

Acknowledgements

This work was supported by a grant from The City University of New York PSC-CUNY Research Award Program.