Exploratory Patent Search with Faceted Search and Configurable Entity Mining

Pavlos Fafalios[1], Michail Salampasis[2] and Yannis Tzitzikas[1]

[1]Institute of Computer Science, FORTH-ICS, and
Computer Science Department, University of Crete, Greece
[2]Vienna University of Technology, Institute of Software Technology and Interactive Systems
Emails: fafalios@ics.forth.gr, salampasis@ifs.tuwien.ac.at, tzitzik@ics.forth.gr

Abstract

Searching for patents is usually a recall-oriented problem and depending on the patent search type, quite often a problem which is characterized by uncertainty and evolution or change of the information need. We propose an exploratory strategy for patent search that exploits the metadata already available in patents in addition to the results of clustering and entity mining that are performed at query time. The results (metadata, clusters and entities grouped in categories) can complement the ranked list of patents produced from the core search engine with useful information for the user (e.g. providing a concise overview of the search results) which are further exploited in a faceted and session-based interaction scheme that allows the users to focus their searches gradually and to change between search methods as their information need is better defined and their understanding of the topic evolves in response to found information. In addition, we propose the exploitation of Linked Data for specifying the entities of interest and for providing further information about the identified entities. The proposed system offers a dynamic, entity-based integration of patent documents, patents metadata and other external (semantic) resources.

1 Introduction

Patent search is a type of professional search and most patent searches (e.g. patentability and validity) are crucially important for businesses' patent management success. Missing relevant documents is unacceptable many times therefore the retrieval of all relevant documents is usually necessary. In that context the provision of facets that correspond to various kinds of metadata or extracted entities can help the user to get an overview and to quickly restrict the search space and help users in making sense of query results in the context of the information retrieved and views produced from metadata. The usefulness of entity mining in patent search is also revealed by the emergence of systems like *quantalyze*[1] in which quantities such as temperatures are spotted in the documents (patent documents in this case), their respective semantic context is identified and the quantity itself is normalized to a standard unit. However, in patent search more kinds of entities should be supported, e.g. companies, countries, product types, drugs, diseases, etc. Since most of them are *named* entities the exploitation of LOD (*Linked Open Data*[2]) is indispensable because a lot of information about named entities is already there. Furthermore, the use of LOD can bring wide coverage and fresh information.

In: M. Lupu, M. Salampasis, N. Fuhr, A. Hanbury, B. Larsen, H. Strindberg (eds.): Proceedings of the Integrating IR technologies for Professional Search Workshop, Moscow, Russia, 24-March-2013, published at http://ceur-ws.org

[1]https://www.quantalyze.com/en/
[2]http://linkeddata.org/

In this paper, we propose an exploratory search system which uses static and dynamically generated metadata and combines the following methods: a) exploiting at query time the *static metadata* of the top results, b) *clustering* and *entity mining* that are performed at query time (no pre-processing or indexing is necessary) for the domain of patent search and the ability to *configure* and personalize the *entities of interest*, and c) exploring the entities and their characteristics by exploiting the LOD cloud. The results (metadata, clusters and entities grouped in categories) can complement the query answers with useful information for the user (offering a concise overview of the search results) which are further exploited in a *faceted* and *long session* interaction scheme that allows users to restrict their focus gradually as their information need is better defined.

We believe the work presented in this paper addresses the issue of integrated patent/professional search systems from two important perspectives. From an *information integration* point of view *entity names* are used as the "glue" for automatically connecting documents (patents in our case) with data (and knowledge). This approach does not require deciding or designing an integrated schema/view, nor mappings between concepts as in knowledge bases, or mappings in the form of queries as in the case of databases. Note also that in professional patent search, in many situations, one must look beyond keywords to find and analyze patents based on a more sophisticated understanding of the patent's content and meaning [JAV10]. Technologies such as entity identification and analysis could become a significant aid to such searches and can be seen, together with other text analysis technologies, as becoming the cutting edge of information retrieval science [BCC10].

From an *information seeking* process perspective we present the tight integration of different search tools for a) faceted search using existing metadata, b) entity extraction and c) textual clustering, with the main retrieval engine which produces ranked lists of patent documents in response to a query. This integration allows different search interfaces to coexist in an information seeker's patent search system and may be seen as a desirable feature, but it could also easily lead to a feeling of "information overload" on the searcher's side. To address this risk the tools are synchronized so one event or action in one tool (for example selecting a facet) can update the views produced from the other tools. All search tools, taken together, could provide professional search systems better supporting exploratory search characterized by recall-oriented information problems. Ultimately, users working within this complex information workplace, should have at their disposal multiple tools, interfaces, and engage in rich and complex interactions to achieve their goals as their understanding of the topic is increased and the information need is better defined.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 describes and analyzes the proposed functionality. Section 4 reports experimental results, and finally, Section 5 concludes and identifies issues that are worth further research.

2 Related Work

Many countries provide web interfaces for searching their patent databases, e.g. the *United States Patent and Trademark Office* (USPTO)[3] and the *European Patent Office* (EPO)[4]. There are also many well-known web systems that retrieve information from several patent databases like *Google Patent Search*[5], *Free Patens Online* (PTO)[6] and *Patents.com*[7], and some commercial systems like *Delphion Derwent*[8] and *Thomson Innovation*[9]. Furthermore, several workshops have been conducted to evaluate and improve the state of the art in patent retrieval [LHR11, HZB10] and several approaches have been proposed. [OSM97] introduced a system that integrates a series of shallow natural language processing techniques into a vector-based document information retrieval system for searching relevant patents, while [Lar99] proposes a probabilistic information retrieval system for searching and classifying US patents. Other works apply text analysis and retrieval methods to improve recall and precision [MMO+05], investigate cluster-based retrieval in the context of invalidity search task of patent retrieval [KNKL07], or construct clusters of patents containing same classification codes and employ cluster based retrieval [KKL06]. [TLL07] introduced a series of text mining techniques for patent analysis, including text segmentation and summary extraction, [HFAS09] uses classification code hierarchy to find similar patents, while [XC09] considers a search scenario in which users can pose full patents as a query. Most recent works focus on increasing the retrievability of patents by expanding prior-art queries generated from query patents using query expansion with pseudo relevance feedback [BR10], or propose a topic-driven patent analysis and mining

[3] http://patft.uspto.gov/
[4] http://www.epo.org/searching.html
[5] https://www.google.com/patents
[6] http://www.freepatentsonline.com/search.html
[7] http://www.patents.com/
[8] http://www.delphion.com/derwent/
[9] http://info.thomsoninnovation.com/

system [TWY+12]. However, to the best of our knowledge there is no work on patent search that performs at real time textual clustering and (configurable) entity mining on the top results returned by a patent search system, exploits at real time semantic repositories and offers a faceted search-like interface.

The idea of enriching the classical *query-and-response* process of current web search engines, with *static* and *dynamic* metadata for supporting *exploratory search* was proposed in [PKA09] and it is described in more detail (enriched with the results of a user-based evaluation) in [PAKT12]. In that work the notion of *dynamic metadata* refers to the outcome of *results clustering* algorithms which take as input the *snippets* of hits, where snippets are *query word dependent* (and thus they cannot be extracted, stored and indexed a-priori). Note that the results of entity mining if applied over the textual snippets also falls into the case of *dynamic metadata*.

[FKM+12] presents a method to enrich the classical (keyword based) web searching with *entity mining* that is performed at query time. In addition, it shows how Linked Data can be exploited for specifying the entities of interest and for providing further information about the identified entities. That work showed that the application of entity mining over the snippets of the top hits of the answers can be performed at real-time. Mining over the full content of the top results returns much more entities but is very time and memory consuming. Our intention is to apply the aforementioned techniques plus the results of textual clustering and metadata-based grouping in *patent search*, with focus on the needs of patent searchers.

Regarding the value of the supplementary information to users in web searching (in our case the metadata groupings, the clusters and the entities), experimental results have shown that categorizing the search results improves the search speed and increases the accuracy of the selected results [KA05]. Moreover, the user study in [Käk05] showed that categories are successfully used as part of users' search habits. Specifically, users are able to access results located far down in the ranked order list and formulate simpler queries in order to find the needed results. In addition, the categories are beneficial when more than one result is needed like in an exploratory or undirected search task. A user study in [PF00] indicated that categorizing the results dynamically in a medical search system offers an organization of the results that is more clear, easy to use, accurate, precise, and helpful than the simple relevance ranking. Finally, a study in [AJK05] showed that experienced web users prefer to use clustering when they are trying to get an overview or explore a topic.

3 Real-time Exploratory Patent Search

We focus on a *dynamic* approach where no pre-processing of the resources has been done. Specifically, the user submits a *keyword query* and the system fetches the *top-L results* (i.e. the potentially most relevant patents) from the underlying search system, including the static metadata of each patent. Then we apply *textual clustering* and *entity mining* in the title and abstract of the top-L results. Afterwards, we *group* the hits in categories according to their static metadata, clusters and entities and *rank* the - often numerous - elements of each category.

Figure 1 depicts an indicative screendump of a prototype system that we have designed and developed (we analyze it more at Section 3.8). Note that the user has many options for restricting the search space by selecting one or more *entities* (**A**), *metadata values* (**B**), or *clusters* (**C**). For instance, in the current example the user has focused to the Drug *ibuprofen*, the International Patent Classification *A61531/185*, and the Cluster *behandlung* (**D**), restricting the search space to only 2 results. Furthermore, he is able to retrieve at real-time more information about an entity (**E**). We analyze more the proposed interaction model in Section 3.6.

3.1 Metadata-based grouping

For grouping the results according to their static metadata values, we had to discover which metadata elements are important in a patent search. Specifically, we need to know which fields are most useful and likely to be used in a faceted search-like interface for narrowing the search space. Note that a patent document may have numerous metadata elements. For instance, a patent document of the *Matrixware Research Collection* (`MAREC`) data corpus[10] may contain more than 20 metadata elements including *document identification fields, concerned parties, filing and priority information, national and international classification codes, titles, abstracts and descriptions (in many languages), citations, related applications, claims*, etc. For selecting the fields which are most useful and likely to be used in a faceted search-like interface, we gathered opinions during interviews with patent examiners from the *Industrial Property Organization of Greece*[11], in a visit aiming to observe patent examiners searching in their working environment. Later on we did an one-on-one interview with a very experienced patent examiner specifically to learn about their attitudes and beliefs surrounding the usefulness of different types of metadata in patent search. The expert mentioned us the following 9 metadata fields as being important in a faceted

[10] http://www.ir-facility.org/prototypes/marec
[11] http://www.obi.gr/

Figure 1: A prototype offering real-time exploratory patent search.

patent search: *International Patent Classification (IPC), European Classification (ECLA), Applicant, Inventor, publication number, publication country, publication year, application country, application year.* Thus, we decided to offer the above metadata fields for faceted exploration.

3.2 Textual Clustering

Results clustering is very useful for providing users with an overview of the results. It aims at grouping the results into topics (called clusters), with predictive names (labels), aiding the user to locate quickly one or more documents (patents in our case) that otherwise it would be very difficult to find, especially if they are low ranked. In our setting, we use a variation of the Suffix Tree Clustering (STC) algorithm [ZE98], called `NM-STC` (No-Merge STC) [KPT09], that derives hierarchically organized labels and is able to favor occurances in a specific part of the result (e.g. in the title). The last is very useful for clustering the results of a patent search because we want to favor occurrences in the title of the patents, since the *invention title* usually is the most important and descriptive part of a patent.

3.3 Discovering Entities

We currently use `GateAnnie`[12] for entity mining. In our setting it takes as input a set of textual contents (the title and abstract of each patent), specifically those of the top hits of the query answer, and it returns as output a set of entity lists (one list for each category of entities). We have automated the procedure of adding a new category of entities in `GateAnnie`. Thereby, we can easily configure the entity names that are interesting for the application at hand (e.g. LOD of a particular type). As we will see later (Section 3.7), for defining the *entities of interest* we can exploit any semantic repository that is accessible via a SPARQL endpoint, or we can load our own lists of entities.

3.4 Ranking the Entities and the Static Metadata Values

Note that the discovered entities and the static metadata values may be numerous. Thereby, we need an affective method for ranking them and promoting the most important. Consider that the user submits a keyword query q, and let P be the set of the top-L patents (e.g. $L = 200$) returned by the underlying search system. For a $p \in P$, let $rank(p)$ be its position in the answer (the first result has rank equal to 1, the second 2, and so on). Regarding

[12] http://gate.ac.uk/ie/annie.html

the ranking of entities, we apply entity mining in P, get a set of entities E, and rank each $e \in E$ according to the formula: $Score(e) = \frac{\sum_{p \in pats(e)} ((|P|+1)-rank(p))}{\frac{|P|(|P|+1)}{2}}$, where $pats(e)$ denote the patents (i.e. the elements of P) in which an entity e has been identified. We can see that the entities occurring in the top results are promoted, i.e. we exploit the ranking of the patents. The rational behind this ranking formula is that the top patents in the ranked list probably contain more useful entities that the last patents since they are considered "better" results. On this account (and considering that we analyze only the top-L results), the ranking algorithm of the underlying search system is very important.

We use the same formula for ranking also the static metadata values in a metadata category, e.g. for ranking the applicants, the publication years, etc.

3.5 Ranking of Metadata and Entity Categories

For a category c, if $inst(c)$ denotes the entities or the metadata values that fall in c, we rank the categories according to the formula: $Score(c) = \sum_{e \in inst(c)} Score(e)$. We can see that the categories which contain the more highly scored entities or metadata values are promoted.

3.6 The Proposed Interaction Model

Faceted search-like exploration of the (top) results

The results of entity mining, clustering and metadata-based grouping are visualized and exploited according to the *faceted exploration* interaction paradigm [ST09]: when the user clicks on an entity, a cluster or a metadata value, the hits are restricted to those that contain that entity, cluster or metadata value (Figure 1D). Specifically, the user is able to gradually select elements from one or more categories and refine the answer set accordingly (the mechanism is session-based). If such selections belong to the same category, they have disjunctive (`OR`) semantics and if they belong to separate categories they have conjunctive (`AND`) semantics. Furthermore, the user can see only the top-10 values in each category and by simply clicking a hyperlink (button) he can inspect all of them.

On-click semantic exploration of the Linked Data

There are already vast amounts of structured information published according to the principles of LOD. The availability of such datasets enables not only to configure easily the entity names that are interesting for the application at hand (see Section 3.7), but also the enrichment of the entities with more information about them. In this way the user not only can get *useful information* about one entity without having to submit a new query, but he can also start *browsing* the entities that are linked to that entity. Note that many of the *static metadata* can also be considered entities, e.g. the Applicant, the Inventor, the Publication Country, etc.

Another important point is that exploiting LOD is more dynamic, affordable and feasible, than an approach that requires each search system to keep stored and maintain its own knowledge base of entities and facts. Returning to our setting, a question is which LOD dataset(s) to use. One approach is to identify and specify one or more appropriate dataset(s) for each category of entities. For example, for entities in the category Publication Country, the *GeoNames*[13] dataset seems ideal since it offers rich information about countries. Furthermore, *DBpedia*[14] is appropriate for multiple categories such as Applicants, Countries, Inventors, etc. Other sources that could be used include *FreeBase*[15] (for persons, places and things) and *YAGO* [SKW07] which includes *Wikipedia*, *WordNet* and *GeoNames*. In addition *FactForge* [BKO+11] includes 8 LOD datasets (including *DBpedia*, *Freebase*, *Geonames*, *Wordnet*). Many of the aforementioned datasets offer access through SPARQL endpoints[16].

Running one (SPARQL) query for each entity would be a very expensive task, especially if the system has discovered a lot of entities. For this reason, we offer this service *on demand*. Specifically when the user clicks on the small icon at the right of an entity's name, the system *at that time* collects more information about that entity which are visualized in a popup window as shown in Figure 1E. Then, the user is able to continue browsing by exploring the properties of the related entities. As we will see later, the user is able to define a SPARQL endpoint and a SPARQL template query for each category of entities (see Section 3.7).

[13] http://www.geonames.org/
[14] http://dbpedia.org/
[15] http://www.freebase.com/
[16] DBpedia: http://dbpedia.org/sparql, FactForge: http://www.factforge.net/sparql, YAGO: http://lod2.openlinksw.com/sparql

3.7 Configurability

We give particular emphasis on the configurability of the system. The administrator can specify various parameters of the system through a configuration page (Figure 1F). The most important is the exploitation of the LOD cloud for i) dynamically adding a new category of entities, and ii) defining how to semantically explore the identified entities.

Adding a new category of entities

We are able to *add a new category* of entities by giving a *category title* and a *list of words/phrases*. The list can be loaded by running a SPARQL query over a knowledge base that offers a SPARQL endpoint. For example, we can run a SPARQL query over DBpedia's SPARQL endpoint that returns a list of all objects of `rdf:type dbpedia-owl:ChemicalCompound` and thereby offer the ability to explore Chemical Compounds in the search results.

Specifying the underlying knowledge bases

We are able to define how to *semantically explore* an identified entity by giving a *SPARQL template query* and a *SPARQL endpoint* for each category of entities that we want to offer entity exploration. The SPARQL template query must contain the character sequence <ENTITY> (including the < and >). When a user asks for more information about an entity, we read the template query of the category in which the selected entity belongs, and we replace each occurrence of <ENTITY> with the entity's label name.

3.8 The prototype

We have implemented a prototype that offers the aforementioned functionality[17]. The underlying search system searches the `CLEF-IP 2011`[18] data collection which contains more than 2.6 million patent documents extracted from the `MAREC` data corpus. With the current configuration, the system offers faceted exploration of the "important" metadata fields (Section 3.1) and of the entity types Drug, Disease, Chemical Substance and Protein. For exploring the entities, we exploit DBpedia's SPARQL endpoint and we have specified the appropriate SPARQL template queries (all this information can be managed through the configuration page).

4 Experimental Evaluation

4.1 Execution Time

We run 100 random queries in the system described in Section 3.8 and we measured the average time required for a) grouping the top-L results according the their metadata values, b) applying clustering on the title and abstract of the top-L results, c) applying entity mining on the title and abstract of the top-L results, for several values of L. The experiments were carried out using a laptop with processor Intel Core i5 @ 2.4Ghz CPU, 4GB RAM and running Windows 7 (64 bit). The implementation of the system is in Java 1.6 (J2EE platform), using Apache Tomcat 7.

Table 4.1 reports the results. We notice that the metadata-based grouping requires about 0.8 ms per result, the clustering about 3 ms per result, while entity mining is the most time consuming task requiring about 10 ms per result. However, the 3 tasks can be performed in parallel (the results of a task are not required for running another task). Note also that the time does not depend on the underlying data sources but on the number of the top results that we want to analyze; the more results we analyze, the more time is required for grouping, clustering and mining them.

Table 1: Average Time

Number of top results	Metadata-based grouping	Clustering	Entity mining
25	21 ms	79 ms	265 ms
50	38 ms	149 ms	487 ms
100	72 ms	302 ms	1,138 ms
200	149 ms	633 ms	2,229 ms

Note that the user is able to select the number of top-hits, e.g. a bigger L, and thus achieve the desired recall level. However, the higher this number is the more time the system requires. In such cases and for improving scalability one could either build a dedicated index and offer instant response for the most frequent queries according to the approach proposed in [FT11], or adopt a *MapReduce* approach [DG08] for distributing the problem to many computers.

[17] The prototype is accessible through `http://139.91.183.72/x-search-metadata-groupings/`

[18] http://www.ir-facility.org/clef-ip

4.2 Time for exploring an entity

The time for exploring an entity (by querying the LOD cloud) highly depends on the SPARQL endpoint (i.e. the underlying knowledge base) and the SPARQL template query. We have noticed that the more data (i.e. entities) a category of entities contains in the underlying knowledge base, the more time is required for retrieving information. For example (and for the time being), DBpedia's endpoint contains about 5,000 entities of type Drug (`rdf:type dbpedia-owl:Drug`). For retrieving information about a Drug, about 5 seconds are required (including network delay time). However, for retrieving information about a Company (DBpedia contains about 45,000 entities of `rdf:type dbpedia-owl:Company`), the time required is about 20 seconds. Nevertheless, if we know the `URIs` of the entities in a category (and keep them in an index) the retrieval can be performed much faster because the query that we can form is simple and the endpoint will not perform many comparisons since it knows the exact `URI` in which the information lies.

5 Conclusion

We have introduced an exploratory method for patent search that exploits at real time the available metadata plus the results of *textual clustering* and *entity mining* in a *faceted* and *session-based* interaction scheme that allows the user to restrict his focus gradually. We also propose the exploitation of *Linked Data* for specifying the entities of interest and for providing further information about the identified entities. The proposed functionality offers an *entity-based integration* of patent documents, patents metadata and other external (semantic) resources.

In comparison to the existing systems for patent search, the proposed approach offers the ability to a) restrict the focus using static metadata values which are not offered by the advanced search but are important for the patent searchers, b) restrict the focus using entity values and important topics (clusters) that were discovered in the search results, c) exploit any knowledge base that is accessible through a SPARQL endpoint for both retrieving more information about an identified entity and specifying the entities of interest. Furthermore showing values and their count gives an overview (e.g. percentage of patents published in Greece). Note also that the proposed functionality can be exploited by any patent search system (i.e. it acts as a service over a ranked list of results), it does not require any pre-processing and it does not use any caching scheme. The experimental results showed that we can efficiently offer the proposed functionality, however the time that we have to pay is proportional to the number of the top results that we want to "explore". Furthermore, the time for exploring the LOD cloud for retrieving more information about an entity highly depends on the SPARQL endpoint and the SPARQL query that we use.

In future we plan to investigate approaches for entity deduplication and cleaning that are appropriate for our setting. *Entity disambiguation* is a problem that affects the quality of the presented entities and an important issue that worths further research. Ambiguity in an entity name can arise from variations in how an entity may be referenced, e.g. *IBM* and *International Business Machines*, or from the existence of several entities with the same name, e.g. *Argentina (the country)* and *Argentina (the fish)*. Finally, we plan to conduct a user centered, task-based evaluation in order to measure the overall impact of the techniques for structuring patent search result lists.

Acknowledgements

Work done in the context of *MUMIA* (COST action IC1002, 2010-2014) and *iMarine* (FP7 Research Infrastructures 283644, 2011-2014).

References

[AJK05] A. Aula, N. Jhaveri, and M. Käki. Information search and re-access strategies of experienced web users. In *Proceedings of the 14th international conference on World Wide Web*. ACM, 2005.

[BCC10] D. Bonino, A. Ciaramella, and F. Corno. Review of the state-of-the-art in patent information and forthcoming evolutions in intelligent patent informatics. *World Patent Information*, 32(1), 2010.

[BKO+11] B. Bishop, A. Kiryakov, D. Ognyanov, I. Peikov, Z. Tashev, and R. Velkov. Factforge: A fast track to the web of data. *Semantic Web*, 2(2):157–166, 2011.

[BR10] S. Bashir and A. Rauber. Improving retrievability of patents in prior-art search. *Advances in Information Retrieval*, pages 457–470, 2010.

[DG08] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[FKM+12] P. Fafalios, I. Kitsos, Y. Marketakis, C. Baldassarre, M. Salampasis, and Y. Tzitzikas. Web searching with entity mining at query time. In *Proceedings of the 5th Information Retrieval Facility Conference*, July 2012.

[FT11] P. Fafalios and Y. Tzitzikas. Exploiting available memory and disk for scalable instant overview search. *Web Information System Engineering*, pages 101–115, 2011.

[HFAS09] C.G. Harris, S. Foster, R. Arens, and P. Srinivasan. On the role of classification in patent invalidity searches. In *Proceedings of the 2nd international workshop on Patent information retrieval*. ACM, 2009.

[HZB10] Allan Hanbury, Veronika Zenz, and Helmut Berger. 1st international workshop on advances in patent information retrieval. *SIGIR Forum*, 44(1), 2010.

[JAV10] H. Joho, L.A. Azzopardi, and W. Vanderbauwhede. A survey of patent users: an analysis of tasks, behavior, search functionality and system requirements. In *Procs of the 3rd symposium on Information interaction in context*. ACM, 2010.

[KA05] M. Käki and A. Aula. Findex: improving search result use through automatic filtering categories. *Interacting with Computers*, 17(2):187–206, 2005.

[Käk05] M. Käki. Findex: search result categories help users when document ranking fails. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2005.

[KKL06] J. Kim, I.S. Kang, and J.H. Lee. Cluster-based patent retrieval using international patent classification system. *Computer Processing of Oriental Languages. Beyond the Orient: The Research Challenges Ahead*, pages 205–212, 2006.

[KNKL07] I.S. Kang, S.H. Na, J. Kim, and J.H. Lee. Cluster-based patent retrieval. *Information processing & management*, 43(5):1173–1182, 2007.

[KPT09] S. Kopidaki, P. Papadakos, and Y. Tzitzikas. Stc+ and nm-stc: Two novel online results clustering methods for web searching. *Web Information Systems Engineering*, 2009.

[Lar99] L.S. Larkey. A patent search and classification system. In *Proceedings of the fourth ACM conference on Digital libraries*, volume 11, 1999.

[LHR11] Mihai Lupu, Allan Hanbury, and Andreas Rauber. 4th international workshop on patent information retrieval. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, New York, NY, USA, 2011. ACM.

[MMO+05] H. Mase, T. Matsubayashi, Y. Ogawa, M. Iwayama, and T. Oshio. Proposal of two-stage patent retrieval method considering the claim structure. *ACM Transactions on Asian Language Information Processing*, 4(2):190–206, 2005.

[OSM97] M. Osborn, T. Strzalkowski, and M. Marinescu. Evaluating document retrieval in patent database: a preliminary report. In *Proceedings of the sixth international conference on Information and knowledge management*. ACM, 1997.

[PAKT12] Panagiotis Papadakos, Nikos Armenatzoglou, Stella Kopidaki, and Yannis Tzitzikas. On exploiting static and dynamically mined metadata for exploratory web searching. *Knowledge and Information Systems*, 30:493–525, 2012.

[PF00] W. Pratt and L. Fagan. The usefulness of dynamically categorizing search results. *Journal of the American Medical Informatics Association*, 7(6):605–617, 2000.

[PKA09] P. Papadakos, S. Kopidaki, and Y. Armenatzoglou, N.and Tzitzikas. Exploratory web searching with dynamic taxonomies and results clustering. In *Proceedings of the 13th European Conference on Digital Libraries*, September 2009.

[SKW07] F.M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th World Wide Web conference*, 2007.

[ST09] G.M. Sacco and Y. Tzitzikas. *Dynamic taxonomies and faceted search: theory, practice, and experience*, volume 25. Springer-Verlag New York Inc, 2009.

[TLL07] Y.H. Tseng, C.J. Lin, and Y.I. Lin. Text mining techniques for patent analysis. *Information Processing & Management*, 43(5):1216–1247, 2007.

[TWY+12] J. Tang, B. Wang, Y. Yang, P. Hu, Y. Zhao, X. Yan, B. Gao, M. Huang, P. Xu, W. Li, et al. Patentminer: topic-driven patent analysis and mining. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012.

[XC09] X. Xue and W.B. Croft. Transforming patents into prior-art queries. In *Proceedings of the 32nd international ACM SIGIR conference on research and development in information retrieval*. ACM, 2009.

[ZE98] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval*. ACM, 1998.