A Flexible tool for Cross-Collection Patent Search

Stefania Marrara, Gabriella Pasi

DISCo, Università degli Studi di Milano Bicocca

stefania.marrara,pasi@disco.unimib.it

IIR2015 – Cagliari May 25th

Patent Search: issues and peculiarities

- Patent Information Retrieval (PIR) is a specialized branch of Information Retrieval, which is aimed to support users, often professionals such as patent attorneys or inventors, in retrieving patents that satisfy their information needs

- prior-art retrieval is a crucial application: patent authors require an exhaustive knowledge of all related patents

Patent Search: issues and peculiarities

- Today patents are commonly available: USPTO (United States Patent and Trade Office), EPO (European Patent Office) and WIPO (World Intellectual Property Organization).

- Each collection contains several thousands of patents and continues to grow up year by year -> increasing costs!!!

Patent Search: issues and peculiarities

- Patent Retrieval is also considered a complex challenging task:
  - the vocabulary used in patents is often obscure as it contains a lot of specialized or technical words.
  - Often the obfuscation of content is intentional by writers who wish their patents difficult to retrieve;
  - patents contain an intrinsic structure which often include description, claims or prior-art for instance and can be different in different collections.
  - Finally typical queries in patent retrieval include a huge amount of words, often entire claims.

Patent Search: issues and peculiarities

- Most Patent Search tools available today are collection dependent. The most known, Google Patents [4] and PatentsSearcher [5, 14], are mostly centered on the USPTO collection even if the issue of world-wide patents search is perceived.
- Most approaches presented in the literature, based on keyword extraction or query expansion techniques, proved to produce poor results

A Flexible Query language for XML documents: FleXy

- In previous work we defined a flexible extension of the XQuery Full Text language (FleXy) by introducing flexible constraints on both XML document structure and content.
- A patent search application based on FleXy has been proposed in PatentLight.
- In PatentLight the structure-based constraints of Flexy named below and near, and the content-based flexible constraint around where employed
- In this work we introduce the constraint similar which applies on tag names, and we show how the combination of content-based and structure-based evaluation of results can improve the effectiveness of PatentLight.

Below, near, around

- The constraint below retrieves the fragments of an XML document (in this case a patent) that are closer to the path required by the user's query.

  c/below::t $w_{c,t}$= 1/|desc_arc(c,t)|

- Near retrieves elements that are connected to the context node by any path (not only the descendant relationship), i.e., also ancestor and sibling elements are evaluated.

  c/near::t $w_{c,t}$= 1/|arcs(c,t)|

- Around applies to numerical data
  - its evaluation function is formally defined as the membership function of a fuzzy subset
  - In the patent domain, the constraint around is defined to the aim of analyzing date contents.
  - triangular membership function centered on b

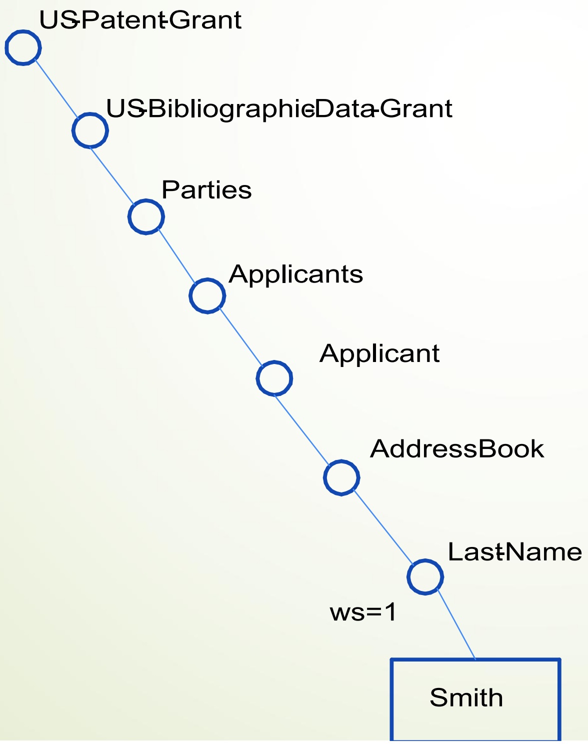
  'tag-date/@date[x around b]

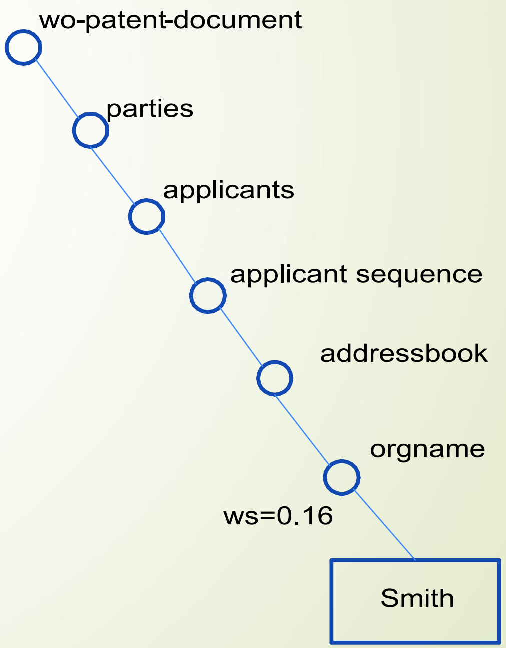
Similar

- Similar is a flexible constraint defined on tag names that allows to retrieve fragments with a target node name similar to the name used in the user query

ws=1/(1+ed) similar(x)

Q1: applicants//similar(LastName)[text() contain text «Smith»]





Query with multiple flexible constraints

- When a query involves more that one flexible constraint, for instance a flexible axis and the similar constraint for the target node name, the overall relevance degree $wo_{c,t}$ is computed as a combination between the two scores, $w_{c,t}$ and ws.

- In principle we prefer a conservative evaluation and therefore we use

  $wo_{c,t} = \min(w_{c,t}, ws)$ but different solutions could be tested.

Using Flexy to query patent collections with heterogeneous structure and tag vocabulary

- PatentLight was previuosly tested on the USPTO patent collection

- it was noted that also EPO and WIPO patent documents show more or less the same structure of USPTO, even if with different tags.

In this paper we use the similar constraint to extend our tests to a cross collections composed by USPTO and WIPO.

Keyword-based approach

- An important functionality of PatentLight is to categorize patents by exploiting their XML structure.
- The engine organizes the XML patents into meaningful semantic XML elements covering the main patent information.
- In this way the categorization process described below can easily capture what the user topical search intent is by identifying the possible interpretations associated with a patent.
- By analyzing the patents in the USPTO collection, four categories were identified in: *People, Title, Description*, and *Claims*.

PatentLight Search

- A user specified keyword based query (here below ”query terms”) is automatically rewritten into four distinct FleXy queries, one for each of the four categories. The structure of each query is predefined in order to search the query terms in pre-established elements as follows:

People:

applicants/near::Last-Name[

text() contains text "query terms"]

Title:

invention-title[

text() contains text "query terms"]

Descriptions:

Description/below::p[

text() contains text "query terms"]

Claims:

claims/below::claim-text[

text() contains text "query terms"]

PatentLight Search

- In this work we improve the PatentLight engine by introducing the possibility to retrieve also fragments with different tag names w.r.t. those expressed by the query.
- In the engine, if the user chose to add the similar tag evaluation the set of FleXy queries would change accordingly as shown below for the category People:

```
People:
similar(applicants)/near::Last-Name[
text() contains text "query terms"]
applicants/near::similar(Last-Name)[
text() contains text "query terms"]
```

PatentLight Search

- The retrieved fragments are ranked according to two values: the degree of structural relevance based on the evaluation of FleXy constraints ($wo_{c,t}$), and the degree of relevance obtained by the full-text scoring of the XQuery Full Text language (the prototype uses the BaseX system).

- The approach privileges the structural ranking w.r.t. the content based relevance since it was observed that the paragraphs most related to the invention are usually structurally closer to the tag Description.

The prototype

- XML patents from the USPTO and WIPO collections published in a small time slot (i.e., from 2015-01-01 to 2015-01-15)
- The final collection is composed by 146.413 XML patents, 82.800 from the WIPO collections and 63.613 from the USPTO collection
- The main module is the BaseX Query engine, which is in charge of the collection indexing process and querying
- One of the main characteristics of the approach is that each query produces a set of results, one for each class (People, Title, Descriptions, Claims), which are not merged.
- The ranking module reorganizes each class of results by first considering wo and next the degree of content-based relevance.

Queries results

	Patent Light				Google Patents
Query	People	Title	Claims	Descriptions	No class
Q1: «Bell»	64(0)	5(1)	98(2)	964(1)	4(0)
Q1.1: «Kettle bell»	0	1(1)	2(2)	3(1)	0
Q2: «gas turbine»	0	215(1)	453(4)	1110(2)	113(2)
Q2.1:«gas turbine compressor»	0	2 (2)	70(2)	341(2)	98(2)
Q3: «Gonzales»	8(1)	0	0	46(1)	40(0)
Q3.1: « Martino Gonzales»	1(1)	0	0	0	0
Q4: «search»	6(0)	279(1)	1770(2)	9487(2)	147(2)
Q4.1: «search engine»	0	25(2)	207(2)	2159(2)	114(0)
Q4.2: «semantic search engine»	0	2(2)	5(2)	139(1)	50(1)
Q5: «transistor»	0	346(1)	2730(1)	8312(1)	199(1)
Q5.1: « low frequency transistor»	0	0	12(4)	910(2)	110(4)

Conclusions and Future work

- In this paper we have described the development and preliminary evaluation of PatentLight on a collection of English patents with dishomogeneous structures.

- The peculiarity of PatentLight is to allow users to specify flexible constraints in their queries.

- Future work will study the evaluation of synonyms for the tag names used in the queries of the four categories.