A Proposed Hybrid Approach for Patent Modeling

Scekic, Ognjen; Popovic, Djordje; and Milutinovic, Veljko

Abstract—***In an effort to find a general model which could capture the essence of any concept, we chose to narrow down the first part of our research to patents only. Patents can be considered as adequate first-step substitutes for concepts in general, because of their diversity and the precision of their descriptions.***

Our aim is to define a model which could allow detection of individual concepts and relationships among them, both within and across patent boundaries.

The approach is based on a hybrid solution – employing existing conceptual indexing techniques for extraction and hierarchical organization of individual concepts, and RDF/OWL descriptions for application-specific data.

1. *Introduction*

EVER since the first computers appeared in the 1940s, people started dreaming of "intelligent machines", capable of not only processing given data, but also understanding it. An important branch of computer science – Artificial Intelligence (AI), is dedicated to achieving that goal. Research efforts in the field of AI provided many useful insights into different new techniques for knowledge extraction and manipulation. Many ideas originating from AI research are being further developed today in a number of related fields. Knowledge representation, Data Mining and Semantic Web are only a couple of those research fields, sharing a common goal – taking data manipulation abilities of a computer to a higher level. We must first make a clear distinction between the terms *data* and *information.* In this context, the term *information* can be defined as a set of relations among different pieces of data that together provide some knowledge. Pieces of data deprived of such relations carry no knowledge.

We will also introduce another term here – concept, to denote a piece of information regarding any concrete or abstract category, as humans perceive it.

In order to take a computer's data manipulation abilities to a higher level, we first need to make data representation and data processing more similar to human's perspective. In order to do that, we need a new layer which could act as an intermediary between a human's perception of reality and a computer's internal way of data representation. Similar aspirations in the field of programming languages led from the initial machine binary language to the creation of object-oriented languages. The missing layer should be able to handle concepts at the level of abstraction as close as humans do. The layer should, basically, be able to deliver two things:

- Define a way to model concepts,
- "Translate" the abstract concepts to a lower level, and vice-versa.

The first point is more of a philosophic and logic nature, whereas the second one is more a technical issue. However, the two issues are closely related, since the complexity and breadth of the first point are inevitably narrowed by the limitations of the second point. The research area that explores these issues is called *concept modeling*.

A good concept model should provide:

- A way of identifying new concepts,
- A way of identifying relations among the new and the existing concepts,
- A way of searching and processing existing concepts to create new concepts/knowledge.

There exist many different models [1] devised for many specific applications. However, the ultimate goal is to find a universal model, powerful enough to capture the essence of any concept.

At IPSI Belgrade we decided to pursue this goal, and devised three different approaches. One of them is to be described in the following part of the paper.

2. *Problem Statement*

In the effort to find a general model which could capture the essence of any concept, we chose to narrow down the first part of our research to patents [5] only. Patents can be considered as adequate first-step substitutes for concepts in general, because of their diversity and the precision of their descriptions. Furthermore, patents have some other very convenient characteristics for this kind of research:

- They are described by a very formal, structured language – claims.

Manuscript received May 30, 2006. This work was supported in part by the Sun Microsystem Company, United States of America.

Scekic Ognjen, Popovic Djordje, and Milutinovic Veljko are with the Faculty of Electrical Engineering, University of Belgrade, Serbia.

- Each patent is a novel concept, defined in terms of existing low-level concepts.
- Each patent is often based on or closely related another one.

Nevertheless, while thinking of a suitable way to model patents we came upon a number of problems:

- How to create a model that has a uniform structure, and can therefore be used for any concept, i.e. to be able to capture the essence of any concept (a patent, in our case)?
- How should these models be linked without creating a myriad of direct links, leading to problems of exponential order when storing and searching such complex structures?
- How could we bridge the gap between natural language and a machine-processable model?

Being a separate problem, we decided to adopt an existing solution to try to solve the last issue – *conceptual indexing* [2, 3, 4] – a method for extracting concepts, word constructs and sentence fragments from any text in natural language, and arranging them in an index.

2.1 Conceptual Indexing

Conceptual indexing is a technique that can improve people's ability to find information in textual materials, using semantic relationships among concepts and natural language processing. This technique is used for indexing and organizing information in structures called conceptual taxonomies that can be used for browsing and information retrieving. The taxonomies represent structured networks of concepts based on conceptual relationships of these concepts.

Conceptual indexing technology can be divided into three major parts that work closely together (Figure 1):

1. Concept extractor – identifies words and phrases to be indexed. It also keeps record of number and places of occurrences of these words.
2. Concept assimilator – analyzes a concept phrase to determine its place in conceptual taxonomy. In other words, it creates the mentioned taxonomy.
3. Conceptual retrieval system – uses conceptual taxonomy to make connections between requested and indexed items. It first uses the concept extractor to identify the requested words and phrases. After that, it uses the Concept assimilator to determine the connections between concept phrases extracted from the query and those placed in taxonomy.

Using the conceptual indexing technique one can relate the terminology of a query to the terminology of some textual information that is placed inside this taxonomy, and conclude with a certain degree of probability that the essence of a text is similar to what is asked in a query. In this way, conceptual indexing can be used to bridge the gap between a natural language and a machine - processable model.

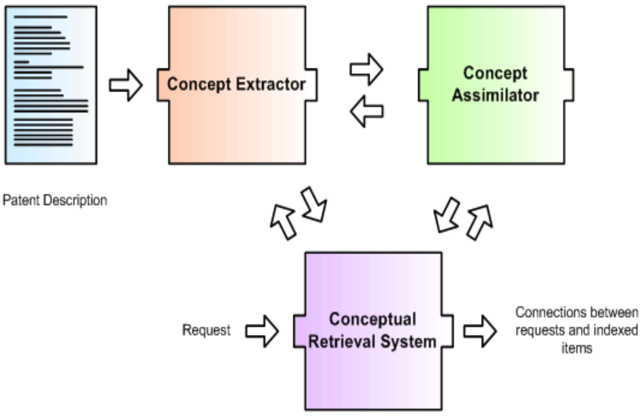
Concept Extractor
Concept Assimilator
Patent Description
Request
Conceptual Retrieval System
Connections between requests and indexed items

Figure 1: Components of a conceptual indexer

2.2 The Seven Ws

The first step of our research was to talk to ordinary people, who are in no way related to this project. We asked them to try to explain how they create a picture of a concept in their heads? The answers were either a number of other concepts that they associate with the concept they were asked to describe, or in the form of answers to some of the 7 Ws (WHAT, WHO, WHEN, WHERE, WHY, WHICH, HOW). (Giving associations can be thought of as giving answers to the WHAT question, so basically it boils down to the same thing.) However, depending on the type of concept, different Ws were used. For example, when describing birds as a class of animals, they would use only certain W associations - WHAT and HOW mostly. But, when they were to describe a historic event, they would use all the 7 Ws. Our conclusion was that only the WHAT associations provided general facts about *any* concept, and were *always* present. Other Ws provided extremely useful additional information when present, but were not always present.

We decided to use a conceptual indexer, to go through the text we want to model into a concept, and create an index of terms, phrases and sentence fragments (later referred to as *terms*). We will use this "small index" – *descriptive index,* (its size is approximately 1-5% of the analyzed text) as a list of WHAT associations. Other Ws will be used depending on the field of application.

For example, when describing patents:

- WHICH – All the numbers that describe a patent (e.g. application number),
- WHEN – All the dates that describe a patent (e.g. when it was filed),
- WHERE – e.g. addresses of the inventors,
- WHO – e.g. inventors, examiners, attorney,
- etc.

As we can see, each of these Ws can have several sub-categories, which are application-specific. Therefore, we could use RDF/OWL [6, 7, 8, 9] statements to capture all this information

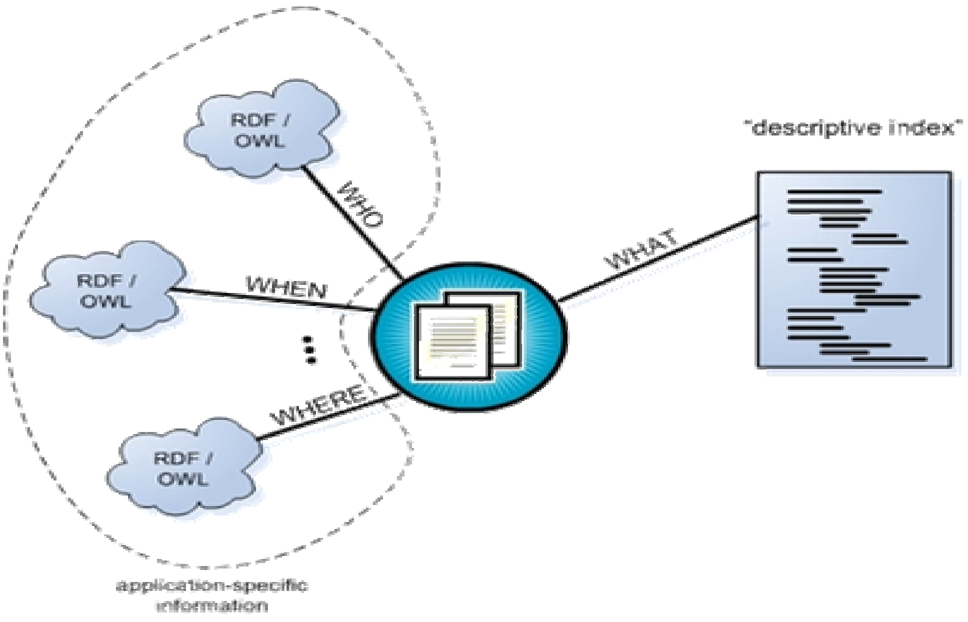

Figure 2: A simplified patent model

properly. This could give us the exactness and the expressiveness like that of ontology, but without driving us into problems of complexity, because we could limit the RDF statements to atomic concepts only. This way, we can easily use automated reasoners to process this data.

3. *Existing Solution and their criticism*

3.1. Ontology

If we wanted to place concepts in an ontology, we would either have to use a well-established ontology (which might not suit our needs), or create a new one. In the latter case, we could create several different ontology/taxonomies/structures, depending on what information we want to capture, and how we want to capture it. However, if we later want to merge our ontology into a different one, we would have to carefully examine existing relationships, determine potential equivalent classes, create new links, etc. This is because in a regular ontology we have all kinds of links, not just between atomic concepts. Such approach works fine within closed community with specific needs, where already exists a well-defined basic ontology structure, and the community members have a good knowledge of how to model new concepts in terms of the existing ones. In such cases, that approach provides a very information-rich model.

In our case, we think it might be better to use only simple relations between atomic concepts. The result will be a loss of much of the expressiveness, but with the benefit of a reduced complexity. This loss is to be compensated through the use of indices.

We hope the research done by the other groups of our team could prove very helpful in this area.

3.2. Indices

For example, let us take the simplest possible definition, for a bird:

Our index might then contain the following associations: creature, wings, feathers, eggs, fly.

Our approach does not offer the possibility to explicitly state the fact that some birds do not fly, as ontology does, but it does allow us to create a simple model, similar to what humans have on their mind when they think of a bird.

Having enough associations, one can create a model with a considerable degree of accuracy. The more associations we have, the clearer picture we get. It is not a question of making an exact guess, but making a near, or very near guess. This is usually good enough for many applications.

It is also important to keep track of how many times a term is mentioned in the text, because it affects its descriptive power. For example, in the claims section of a patent we used, the most frequent terms were “synthetic grass [10]” and “playing surface [9]”. Clearly, these terms represent the essence of what is being described.

	Conceptual indices	**RDF/OWL ontologies**
Major advantages:	Linear-complexity structures	Very expressive and precise
	Provide basic subsumption relations	Based on First-Order Logic
	Provide built-in knowledge on low-level concepts	Supported by W3C
Major drawbacks:	Incapability of establishing explicit relations among high-level concepts	Great complexity
	Incapability to create precise models	

Figure 3: Comparison of the two methods

However, this is only because we *know* what “synthetic”, “grass” and “surface” are. So, at some level, we need to have some intrinsic, built-in knowledge, so that all the other concepts can then be described in terms of these basic concepts. Conceptual indexing could provide us with that built-in basic language understanding.

Basically, our research is a hybrid approach aiming to use advantages of one technique to eliminate the drawback of the other one.

4. Proposed Solution

While the RDF/OWL part of the model is used to link atomic concepts links between more abstract concepts (patents in this case) can be established dynamically, through the use of a joint index. This joint index – *system index*, is created by merging indices of individual concepts, while retaining the links between its terms and their concepts of origin. The merging is done once again by the conceptual indexer, by processing all the descriptive indices to produce the system index. The key advantage here is that adding, removing and searching for concepts is quite easy, and requires little time, because there are no direct, explicit links among them (except among parts that are modeled with RDF/OWL, but those are far less complex issues because we have a predefined, limited structure in such a case).

For example:

When describing two different vaccines we would probably make a frequent use of terms like: vaccine, inactivated antigens, immune response, etc.

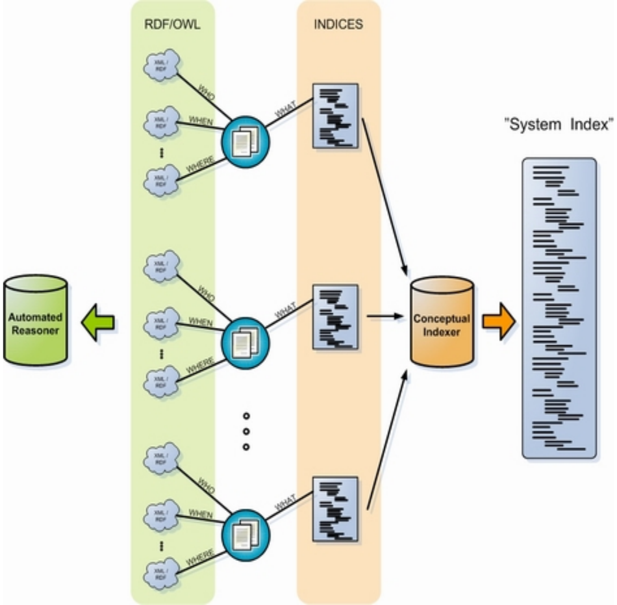

Figure 4: Top-level scheme

By determining overlapping terms we create dynamic, implicit links among similar concepts. Number of such implicit links can be used to express similarity among concepts.

Of course, we need not to have identical terms to establish that there is some level of similarity. Thanks to the existence of basic subsumer-subsumee relations in a conceptual index, we can also infer similarity of patents based on the number of occurrences of similar terms.

The results of the experiment we made on a set of patents referring to communication devices and protocols showed that there is a significant number of overlapping terms. If the number of occurrences of terms is propagated vertically via subsumer-subsumee relations and gradually incremented as we move up the structure, then we get a clear picture of what the patent is actually about. The same experiment also showed that the above-mentioned method does provide a credible way of establishing similarity among patents.

However, the algorithm for establishing similarity can only be tweaked empirically, and its performance may vary according to the field of application. We think the structure of the claims section of a patent document is very appropriate for our approach also because the claims impose a frequent use of key terms.

5. Conclusion

Our idea is still in the first stage of development. Its key advantages are its general applicability and reduced complexity, at the price of reduced precision; a consequence of using indices.

Further research is needed to explore the quality and feasibility of the proposed solution. However, we expect that the combination of OWL/RDF structures and indices might produce a satisfactory performance/exactness ratio.

References

[1] Omerovic, S., Savic, D., Tomazic, S., "A Survey of Concept Modeling,” Faculty of Electrical Engineering, University of Ljubljana, Slovenia (to appear).

[2] Woods, W. A., “Conceptual Indexing: A Better Way to Organize Knowledge,” Technical report, *Sun Microsystems Laboratories*, 1998.

[3] Woods, W. A., Bookman, L. A., Houston, A., Kuhns, R. J., Martin, P., Green, S., "Linguistic Knowledge Can Improve Information Retrieval,” *Proc. of the Applied Natural Language Processing Conference* (ANLP-2000), Seattle, 2000.

[4] Dobrov, B. V., Loukachevitch, N. V., Yudina, T. N., "Conceptual Indexing Using Thematic Representation of Texts,” *Scientific Research Computer Center,* Moscow State University, Moscow, 1998.

[5] http://www.uspto.gov – U.S. Patent Office

[6] Scekic, O., Bojic, P., "An Overview of OWL and its Role in Semantic Web Architecture,” *YU-INFO 06,* Kopaonik, Serbia & Montenegro, 2006.

[7] Klyne, G., Carroll, J., “Resource Description Framework (RDF): Concepts and Abstract Syntax,” *W3C Recommendation*, http://www.w3.org/TR/rdf-schema, 2004.

[8] Hayes, P., “RDF Semantics,” *W3C Recommendation,* http://www.w3.org/TR/rdf-mt, 2004.

[9] Patel-Schneider, P. F., Hayes, P., Horrocks, I., “OWL Web Ontology Language: Semantics and Abstract Syntax,” *W3C Recommendation,* http://www.w3.org/TR/owl-semantics/, 2004.