ONTOLOGY-BASED PRIOR ART SEARCH

Alexey Bondarenok

ScienceSoft, Minsk, Belarus, e-mail: *AlexeyBondarenok@scnsoft.com*

Abstract. This article describes a method of prior art document search based on semantic similarities of a user query and indexed documents. The ontology-based technology of knowledge extraction and representation is used to build document and query images, which are compared using the semantic similarity technique. Documents are ranked according to their semantic similarities to the query, and the top results are shown to the user.

Introduction

Prior art search stands for automatic retrieval of documents (e.g. patents, articles, etc.) that describe already existing problems and solutions similar (or even the same) to the problem or solution described by a user query. Documents are ranked according to their similarities to the query. This feature may be extremely useful for researchers to check if the solution for their problems already contrived, possibly patented, and to find related problems and solutions. The ontology-based technology of knowledge extraction and representation and corresponding semantic similarity calculation technique allow to achieve great results in prior art search.

Ontology

Ontology is a detailed specification of a subject domain, its concepts, their attributes and features, as well as relationships between concepts [1]. Thus, this is a lexicon of terms (concepts) and logical expressions describing the meanings of the terms and their relationships. For every concept C_i its structure, features, set of relationships are extracted and functional and semantic definitions are built.

Concept C_i

- Structure of C_i (main, attribute)
- Functional definitions and features of C_i:
  - C_i is ..
  - C_i includes ..
  - C_i consists of ..
  - C_i has (parameters, advantages, disadvantages, ...)
- Semantic definitions of C_i (tags)

Relations

- Hierarchical
- Related
- Functional
  - SAO relations (SA and AO functions for a concept)
  - Cause-Effect relations

SAO and Cause-Effect relations represent such classic knowledge elements as facts and rules of the subject domain. SAO stands for Subject-Action-Object, Subject performs Action on an Object. For descriptions of hierarchical and related relations see [1].

Ontology-based technology includes Linguistic Processor and Ontology Processor (Fig. 1).

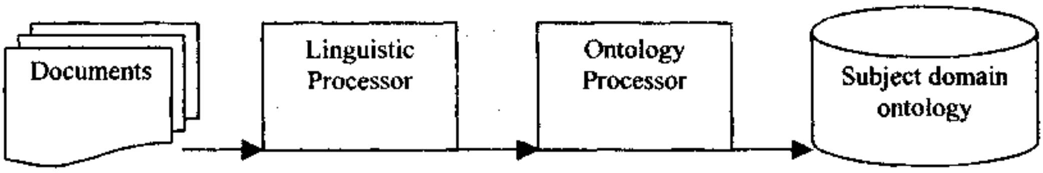

Fig. 1 Ontology creation

Documents of the subject domain are inputted in Linguistic Processor, which performs text preformatting, word and sentence boundary disambiguation, lexical-grammatical, syntactical and semantic analysis. As a result of this processing a set of features and relations of all mentioned language levels is assigned to a set of linguistic units. For instance, there are lexical-grammatical classes of words determined, nominal and verbal syntagmatical sequences, objects and their attributes extracted, etc. Using the results of the linguistic analysis of the text and recognized linguistic models (patterns) Ontology Processor collects the knowledge concerning the subject domain, i.e. builds subject domain ontology, and then represents it in a format appropriate for a further use.

Indexing

Let us assume that we have a set of documents (e.g. patents) and each of them describes a specific problem. To perform search operation within this set of documents a search index should be created. According to ontology-based model each document is processed with Linguistic and Ontology processors and a set of knowledge units that represent the content of the document is extracted. Each knowledge unit belongs to corresponding knowledge category, for example SAO, AO, SA, concept, etc. Each knowledge category represents specific aspect of document content and has a level of importance (weight) assigned. For example, SAO category is more valuable than concept category. It means that one SAO carries more information about document content than one concept does. However, the number of extracted SAOs for a document is usually considerably lower than the number of extracted concepts.

Each knowledge category is independent and therefore can be handled separately from the others, i.e. a separate index can be built and independent search results received. Then search results from different knowledge categories are combined together according to weights of knowledge categories.

For each knowledge unit from any particular knowledge category some frequencies are calculated. There are frequencies of occurrence of the knowledge unit within each document and the frequency of its occurrence within the whole document set.

The following example illustrates several SAOs and concepts extracted from a document; frequencies of occurrence of knowledge units within the document are given in parenthesis.

- SAOs: cover means - cover - spokes(18), spokes - connect - hub(16), bent portion - have - free end(14);
- concepts: hub(180), cover means(97), spoke(89), rim(87), bent portion(80), driver head(72).

So, for each input document its image is built. Document image is a set of knowledge units (with frequencies) extracted from document text and separated by knowledge categories. A set of document images forms the search index that is stored on hard drive in fast-access format.

There are actually several independent parts every document image consists of (knowledge categories). Moreover, document images are independent from each other. So, the search index can be separated into several independent parts, and indexing operations and then search operations can be performed concurrently for them.

Search

The search index is ready to use. Let us examine how the search is committed and semantic similarity is evaluated. The user submits query that describes a particular problem. The query is treated as a single document, i.e. it goes to Linguistic and Ontology processors and the query image is built in the same way as document images were built on indexing stage. To gain more precision user have to describe his problem in details, or in other words, query image should not be too small.

Then the query image is compared with every document image stored in the index. The comparison of the query image A and the document image B includes the following steps:

For each knowledge category K_i, $i \in 1..M$, M is the number of knowledge categories, determine a set of common knowledge units in this category for the compared images, i.e. build an intersection I_i;

For each common knowledge unit $U_{ij} \in I_i$, $j \in 1..|I_i|$ add corresponding bonus B_{ij} to the similarity S_i of the images by selected knowledge category K_i;

Evaluate the final semantic similarity of document images S by combining the results from all M categories.

The bonus B_{ij} depends on normalized frequencies of occurrence of the knowledge unit U_{ij} within each of two images, and on the frequency of the knowledge unit occurrence within the whole indexed document set. Mentioned normalized frequencies are evaluated by dividing the frequency of the knowledge unit occurrence in an image by total frequency of all image's knowledge units in this category. The greater the normalized frequencies are (reliability factor) and the closer they are to each other (nearness factor), the greater bonus will be added. The character factor is the originality of the knowledge unit for the two selected images comparing to the whole document set. This factor is used for slight correction of the bonus. So, the bonus is computed from the following formula:

$$B_{ij} = R_{ij} \cdot N_{ij} \cdot C_{ij}, \tag{1}$$

where R_{ij} – reliability factor, N_{ij} – nearness factor, C_{ij} – character factor.

A wide range of functions was tested to represent each of reliability, nearness and character factors. For example, the following functions may be used:

$$R_{ij} = \sqrt{\nu_{Aij} \cdot \nu_{Bij}}\ , \tag{2}$$

$$N_{ij} = 1 - \left|(1-\nu_{Aij})^2 - (1-\nu_{Bij})^2\right|, \tag{3}$$

$$C_{ij} = 1 + \frac{f_{Aij} + f_{Bij}}{F_{ij}}, \tag{4}$$

where f_{Aij} and f_{Bij} are frequencies of occurrences of the knowledge unit U$_{ij}$ in images A and B, F_{ij} is the frequency of occurrence of the knowledge unit U$_{ij}$ in the whole document set, v_{Aij} and v_{Bij} are normalized frequencies:

$$v_{Aij} = \frac{f_{Aij}}{\sum_{k=1}^{N_{Ai}} f_{Aik}}, \tag{5}$$

$$v_{Bij} = \frac{f_{Bij}}{\sum_{k=1}^{N_{Bi}} f_{Bik}}, \tag{6}$$

N_{Ai} is a size (an amount of knowledge units) of the query image A in the knowledge category K$_i$, and N_{Bi} is a size of the document image B in the knowledge category K$_i$.

The similarity S_i of the images by selected knowledge category K$_i$ is computed from the following formula:

$$S_i = \sum_{j=1}^{|J_i|} B_{ij} \ . \tag{7}$$

As mentioned above the influence of the character factor is quite slight and if we ignore this factor (i.e. $C_{ij} = 1$) then the similarity S_i will be in the range of [0, 1]. Very similar images will get value S_i closer to 1, but completely different images will get S_i equal to 0.

The final semantic similarity of the query and document images S is a weighted sum of similarities S_i:

$$S = \sum_{i=1}^{M} W_i \cdot S_i \ , \tag{8}$$

where W_i, $i \in 1..M$ is an importance of a knowledge category K$_i$.

The user query is compared with every indexed document and corresponding semantic similarity values are evaluated. Documents that are more relevant get greater similarity values. According to these values, documents are ranked in descending order and some of the top results are shown to the user.

Conclusion

Prior art search is a useful tool for researchers to find existing problems and solutions similar to their current problems or topics of research. The described method of prior art search have a great quality (according to response of test users) due to used ontology-based technology of knowledge extraction and representation, and due to the proposed technique of semantic similarity calculation. This approach may be also successfully used in another tasks such as automatic finding of synonymous or related terms to the term given, finding of related documents to the given one, building of proximity matrix when clustering a set of documents, etc. However, the method demands quiet a lot of calculations both on indexing and search stages. Calculations are independent and distributing them among several or more computers in a local network may solve this problem.

References

[1] Sovpel I.V. Ontology-based Technology of Knowledge recognition, extraction and representation. – Informational Systems and Technologies (IST'2002): Proceedings of the I International conference (Minsk, Nov. 5-8, 2002), vol.1, pp.96-99