8

Application of semantic networks in natural language issues

Wojciech Górka, Adam Piasecki and Łukasz Bownik
Centre EMAG
Poland

1. Introduction

Semantic networks are becoming a more and more popular issue these days. This popularity is mostly related to the idea of the so called Web 3.0. However, the use of ontologies and semantic networks is not limited to the Internet. They can find application in data integration, formal description of a domain, identification of facts, etc. Semantic networks are related to natural language applications.

Natural language analysis is based on understanding the user's question and generating an answer. Within the analysis of the user's question there are solutions based both on full-text analysis and on patterns. Full-text analysis is related mostly to Internet browsers or ready-to-use tools which perform such functions. On the other hand, there are solutions based on question patters developed for chatterbot applications. Semantic networks can provide extra qualities to both these solutions, i.e. the possibility to define hierarchies, dependencies between concepts, which will allow the data search to become a more intelligent process.

Semantic networks make it possible to record certain facts and data related by concepts which give meaning to these facts and data. It is especially evident now with the development of data publication (Linked data) in social network services. The advantages of such knowledge collecting processes are: easy navigation between particular concepts, browsing the data in a cross-sectional manner, flexible data structure, and the possibility to record information about meta-data (data structure). On the other hand, there are situations when the information about a particular element should be available as text – sometimes the text is more understandable and readable for the user than a table or structure. Semantic networks, equipped with tools suitable for a given language, easily enable such functionality.

The first section of this chapter will describe semantic network issues. Then two sample solutions will be shown, which use semantic networks for natural language analysis and for generating texts on the basis of data recorded in semantic networks. In the first section the semantic networks issues will be described. Next, two examples of the adaptation of semantic networks in a natural language will be proposed: search engine and natural language generation engine based on semantic networks. The examples will be based on works and tests performed with the use of the Polish language. Still, it seems that the presented ideas will find applications in other languages too.

1.1 Semantic network

The semantic network concept was introduced as an answer to new requirements connected with the progress of the Internet network (Berners-Lee, 2001). The functionality of the Internet (share files, contents, websites, services made available through a variety of forms) is gradually becoming insufficient. Shared resources are primarily intended for use directly by humans. Poor standardization of contents makes it impossible to precise search and process data in an automated manner. For example, e-mail addresses, contact information, calendar of events on a web page are readable for humans. However, if it had to be automatically imported into the mail, calendar, etc. this will be confusing. So it became necessary to build a formalized standard for describing data, knowledge and relationships between them. Formally described data could be both human readable and easily accessible to programs operating on them. A standardized form of data storage will allow to use them in different systems, applications, etc.

1.1.1 Standards related to semantic networks

World Wide Web Consortium (W3C) started to process a knowledge description standard. In 1997 a standard was proposed, and as early as in 1999 W3C published the Resource Description Framework (RDF) standard[1] . The standard was complemented in 2004 with the RDF Schema (RDF-S) (Brickley, 2004) specification.
RDF allows to record triples of concepts. Each triple is a subject-predicate-object expression. Such a way of concepts recording forms a network of definitions (each object can be a subject in a different triple). RDF-S introduced the possibility to build meta-concepts: classes, sub-classes, features. It also launches a non-standard way of defining the name of the notion (label) and its description (comment).
The next stage to extend the semantic web standards was to increase the expressiveness of languages intended for ontology recording. W3C published the OWL (Web Ontology Language) standard (McGuiness & Harmelen, 2004). The language allows, among others, to express the number of concept sets, to show how one concept belongs to or differs from the other, to identify necessary and sufficient conditions for a given concept. Greater expressiveness of the language allows to verify concepts added to the ontology and to search out certain facts and features indirectly. Additionally, OWL makes it possible to integrate two ontologies by means of associating their identical concepts.

1.1.2 Defined ontologies until now

Standards (defined and well known ontologies) allow describing the concepts and connections between concepts. These standards are currently creating a base for the specific schema-ontologies which introduce certain aspects of reality. Sample ontologies:

- Dublin Core (DC) [2]– ontology defining the schema for describing library collections such as books, photos, videos and other multimedia resources;
- Friend of a Friend (FOAF) [3]– ontology which describes the person and the friends of that person, thereby creating a network of connected people;

[1] http://www.w3.org/RDF/
[2] http://dublincore.org/documents/dcmi-terms/
[3] http://www.foaf-project.org/docs/specs

- Semantically-Interlinked Online Communities (SIOC)[4] – ontology which describes social networks;
- DBPedia [5] – ontology which provides data from Wikipedia in the structural way;
- OpenCyc[6] – ontology describing the data collected within the Cyc project. The project aims at mapping the concepts found in the real world and the relationships between them.

Currently, there is also an initiative aimed at linking together different ontologies (Berners-Lee, 2006). This initiative is led by W3C SWEO Linking Open Data. Its purpose is to provide infrastructure for publishing data by means of semantic techniques.

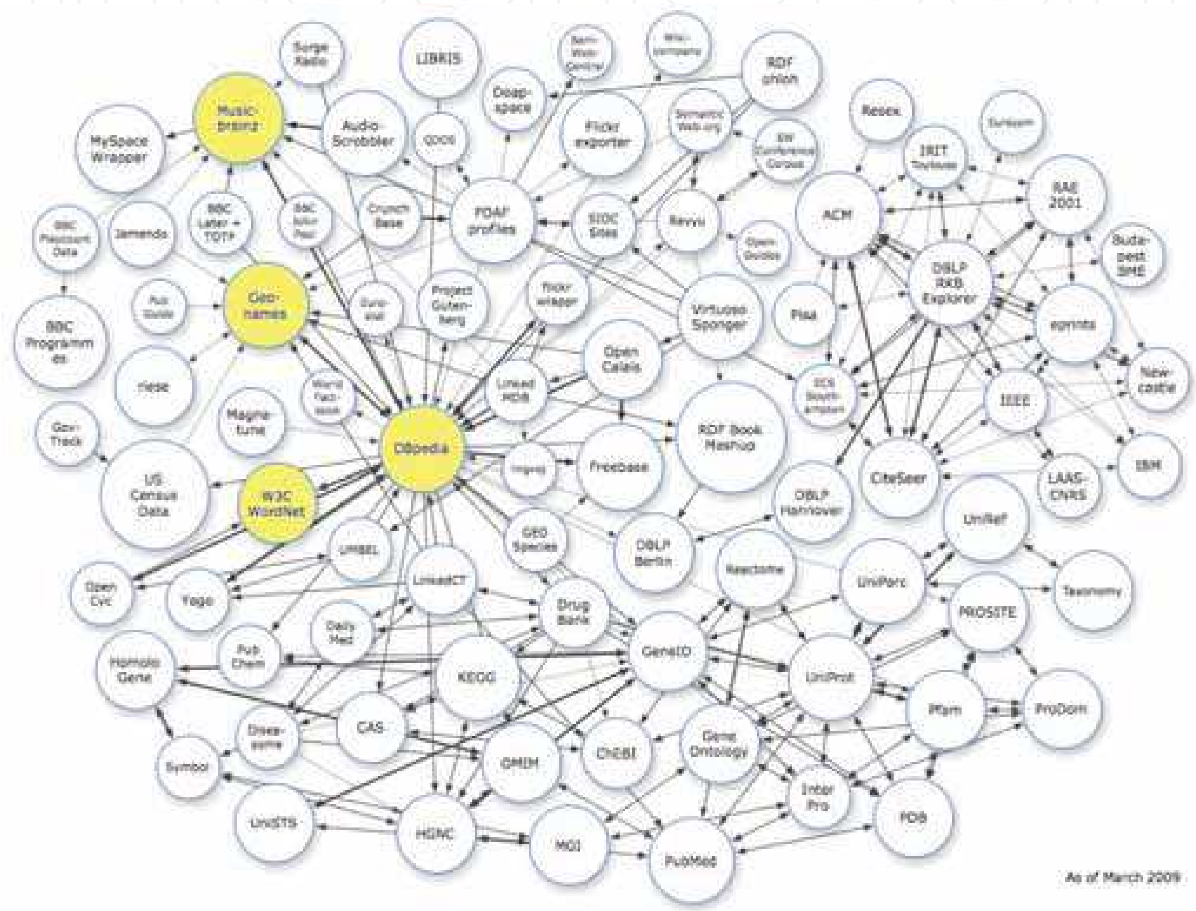

Fig. 1. Existing ontologies and connections between them [source:Linking Open Data Community[7]].

This initiative is primarily concerned with ensuring consistency between existing ontologies (and, therefore, data they describe), so that there would be a smooth transition between the information from different knowledge bases ex. Wikipedia - DBPedia, WordNet, MusicBraintz, Geonames (Figure 1).

4 http://sioc-project.org/ontology
5 http://dbpedia.org/About
6 http://www.cyc.com/cyc/opencyc/overview
7 http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

1.2 Semantic network use cases

Although the idea of semantic networks has been mainly in providing interoperability in the Internet, the work associated with it is also applied in related issues. Currently, semantic networks are seen in the following aspects (Bruijn, 2003):

- as an integrated network of data with different formats;
- as a standard that enables data to define the interface between different fields. As a result, at the intersection of different fields new applications can be produced, benefitting from the recognition of a multi-dimensional issue;
- to support exchange, data sharing, and cooperation on the basis of the same data.

The following areas of application of the semantic network can be highlighted (Saab, 2006):

- linking data with applications (inserting data on a web page and possibility to automatically use them by means of different applications such as: calendar, email, phone etc.) This is due to the standardization of metadata and the implementation of a variety of browser plug-ins that "understand" the data stored in the contents of web pages (e.g. email address, url links, phone numbers, etc.);
- to facilitate filling in the forms. Using ontologies can help to understand the meaning of individual fields in the form;
- combining and integrating data from different sources - the replacement of manual data integration from multiple sources (W3C, 2001). Creating a data bus, the release of data from the application, which allows to create new functionality and easy integration of new systems with the existing ones;
- supporting human cooperation and knowledge acquisition (Davies et al., 2003). Semantic networks facilitate the organization of saving and retrieving knowledge. A sample scenario is the knowledge which is collected by people involved in production and supervision in a factory, etc. They gain knowledge, make decisions, gain experience. However, if it is not recorded, the employees leave it only for themselves. If they do not justify their decisions, a part of their work could be useless in the case they change their jobs. Additionally, in the case of a group of people working on an issue, it is necessary to support the process of saving knowledge, decisions and their justifications. It also permits to track the progress and planning of projects. Other examples of the semantic network application in this area may be as follows:
  - the development of the log of the decisions taken at the stage of production, treatment, evaluation of some facts;
  - maintaining consistency of documentation and informing the service networks about faults;
  - biology, genetics, describing genes, genomes, classification, etc.;
  - description of images and their fragments;
  - customizing therapies with respect to particular patients on the basis of the experience with other cases;
  - integration of research data – different types of data, the structure of the record in the form of troika is more flexible, it is easier to find some "contacts" of data and to view the data multidimensionally;
  - warning about the dangers based on conditions, rules.

- use of semantic networks in the natural language processing (Zaihrayeu et al., 2007), (Jupp et al. 2008);
- integration of geographical data – integration of different data formats, differences in formats and their integration can be assisted by RDF.

As semantic networks are related largely to information processing and organizing, they are also inevitably linked to the issue of natural language. Semantic networks can be used in understanding the text, classification of documents, interpretation of the user's expressions, or to generate dynamic information from the gathered knowledge base. The applications of semantic networks, connected with understanding the users' question and the natural language generation will be described in next sections. The works related to these issues were conducted in the Virtual Consultant for Public Services (WKUP) project, whose objective was to build a service that allows the users to obtain information in the field of competences of the system by means of natural-language communication with the system.

2. Search engine based on semantic networks

The communication with the user is mainly based on understanding the user's question. The issue of interpreting the question can be achieved by text searching as in the case of full-text search engines. On the other hand, there are solutions based on templates related to the user's questions. In the beginning, the existing solutions will be presented. Then, the solution based on semantic networks and ontology Simple Knowledge Organization System (SKOS)[8] will be presented.

2.1 Searching data solutions

In the field of search engines there are widely used solutions as well as those that are not yet applied in production. This section will present the different search solutions available on the market and those which are at the stage of experiments.

2.1.1 Full text search engines

There are many solutions in the realm of information search which allow to index the information contents and search for documents based on the contents. The full-text search solutions are mostly based on statistics and there have been many algorithms developed in order to standardize the search results (Salton & Buckley, 1987). Relatively new solutions are algorithms which allow clustering search results (Manning et al., 2007). The clusterization introduces documents selection with respect to areas of interest (a sort of categorization) based on words used in a given text. A category is, to certain extent, a representation of the document contents determined on the basis of the statistics of words used in the document. Examples of such solutions are Vivismo[9] and Carrot2[10] .

---

[8] http://www.w3.org/2004/02/skos/
[9] http://vivisimo.com/
[10] http://www.carrot2.org/

One of the full-text search products is the Lucene[11] search engine. The engine enables to create a properly compressed index and to efficiently search for documents (even concrete places in documents) which are the answer to the question asked by the user. Additionally, Lucene makes it possible to create adapters which allow browsing different types of documents (Microsoft Office documents, XML documents, PDF documents, etc.)

2.1.2 Solutions based on semantic networks

The application of semantic networks solutions also contributes to improving the search results. Semantic networks allow to describe information in a formal way and to introduce interdependencies between particular pieces of information. This way the information search is broader. The use of semantic webs will allow the search tools developers to design new-quality products. The search tools, equipped with the knowledge about the concepts hierarchy and their interdependencies, will make an impression of intelligent software. Such knowledge allows searching not only for the key words given by the user but also for the related concepts, and shows how this relation is made. On the market, there are search engines which use semantic networks, or at least build results based on the hierarchy of concepts (Hakia[12], Google[13])

2.1.3 Solutions based on language corpora

Irrespective of the development of information technologies, there are works carried out in the realm of text corpora which enable to determine, among others, dependencies between words and the frequency of their occurrence in texts (Przepiórkowski, 2005).
Such works allow creating word nets (WordNet[14]). The works on the word net for the English language have been carried out since 1985. The works on other European languages (Czech, Danish, German, Spanish, Italian, French, Estonian) were carried out between 1996-1999 within the EuroWordNet[15] project.
In Poland the works have been conducted within the plWordNet[16] project. Constructing a word net is done automatically, to a certain extent, thanks to the use of the Polish text corpus. The data from word nets, actually – relations between words, can be used to associate the words which appear in the indexed texts. This way it is possible for the user to find documents on the basis of the question in which the key words included in the document have not been used directly. Thus this solution is similar to proposals derived from the semantic webs concept.
In the realm of information search it is possible to determine the qualities of systems whose objective is to answer the questions. An example is the AnswerBus[17] system based on the knowledge indexed by Internet search tools. The search results are interpreted in an

[11] http://lucene.apache.org
[12] http://www.hakia.com
[13] http://www.google.com
[14] http://wordnet.princeton.edu
[15] http://www.illc.uva.nl/EuroWordNet
[16] http://www.plwordnet.pwr.wroc.pl
[17] http://www.answerbus.com

adequate way so that the information looked for by the user could be extracted from the document found by the search tool.

2.1.4 Solutions based on questions templates

The issue how to interpret the user's questions and conduct a dialogue with him/her was a motive to introduce the AIML[18] language. This language makes the way to create solutions enabling conversation with templates based on questions and answers. The AIML language allows to define the templates of the questions asked by the users. The response is generated based on the found templates. AIML enables to reach simple context dialogues, to store personal information about the user with whom the conversation is processed.
This solution, in spite of various constructions which support the management of templates, seems to be difficult to maintain in the context of a large number of predefined templates.

2.2 SKOS ontology

The Simple Knowledge Organization System (SKOS)[19] the specification developed and extended under the auspices of W3C, defines an ontology which allows to express the basic structure and contents of concept diagrams, including thesauruses, thematic lists, heading lists, taxonomies, terminologies, glossaries, and other kinds of controlled dictionaries. The specification is divided into three parts:

- SKOS-Core - defines basic concepts and relations which enable to develop concepts and relations between them;
- SKOS-Mapping - introduces relations which allow to describe similarities between concepts created in different ontologies;
- SKOS-Extensions - introduces extensions of the intensity of hierarchical relations from SKOS-Core.

The SKOS ontology assumes describing "Concepts". Each "Concept" can be labelled. The SKOS ontology extends (compared to RDF-S) labels that can be used:

- prefLabel (chief label of a given concept);
- altLabel (auxiliary label, alternative for a given concept);
- hiddenLabel (hidden label, e.g. for casual words or other words treated as "hidden" due to other reasons).

The concepts can be linked into hierarchies by means of broader and narrower relations. For example the "Car" concept is broader than the "Van" concept. The SKOS-Extensions specification introduces extra semantics of hierarchy relations, among others by the following relations:

- broaderInstantive / narrowerInstantive (express context hierarchies – instances, e.g. Dog and Azorek[20]);
- relatedPartOf / relatedHasPart (express the whole-part semantics, e.g. Car and Wheel).

---

[18] http://alicebot.blogspot.com
[19] http://www.w3.org/2004/02/skos/, http://www.w3.org/TR/2005/WD-swbp-skos-core-guide-20051102/
[20] Popular dog name in Poland

The SKOS ontology also provides the class definition which describes a set of concepts – Collection. Such a set can help to manage the ontology and facilitate its edition by grouping concepts of similar meanings. Possible ways to use the structures of concepts built on the basis of the SKOS ontology were described in use cases (W3C, 2007). What is derived from these use cases is, among others, the application of SKOS to the following:

- to order and formalize the concepts used in a given domain, to search – on the basis on the concepts and a part of relations between them – for resources assigned to the concepts;
- to search for information in different languages (thanks to an easy method of translating labels in the ontology with an unchanged relation structure);
- to label press articles, TV programmes, etc. with key words from a thesaurus recorded in accordance with the SKOS ontology.

The above objectives of the SKOS ontology satisfy, to a large extent, the requirements of the search tool which was build during experiments. Therefore a decision was made to apply this ontology. The application was justified by the possibility to provide the tool with a wide and, at the same time, precise "understanding" of concepts. Thanks to semantics it is possible to record the relations between concepts which, in turn, allows to better interpret the questions. In comparison with the solutions based on AIML language, this solution seems to be more flexible and easier to maintain and managed. It also allows to control and precisely define the search results.

2.3 The applied search algorithm

The use of the SKOS ontology in the built system consists of two stages: edition and production (search tool operations). The way of using the concepts, defined in accordance with the SKOS ontology, with a view to search for certain resources – data – related to these concepts is demonstrated in Figure 2.

At the edition stage (before the system starts) the administrator defines concepts and their mutual relations. Then he/she creates relations of the defined concepts with the data which are to be searched for. The ontologies defined in this manner are used at the search stage (production operations of the system). The user's question is analyzed based on the used concepts. The identified concepts are processed. On the basis of mutual relations between concepts, the best fitting answers of the system are found – the resources the user is looking for.

The analysis algorithm of the user's question was divided into successive stages. The first stage is "cleaning" the user's question from redundant non-alphanumeric signs as well as lemmatization of particular words in the sentence. For the statement prepared in such a way, at the next stage the best-fit concepts are searched for based on their labels (relations prefLabel, altLabel and hiddenLabel). In the case when the found concepts are not related to the resources, the broaderInstantive, broader and relatedPartOf relations are used in order to search the network for the concepts which have certain resources assigned. This allows to find concepts whose meaning is broader than the meaning of concepts used in the sentence.

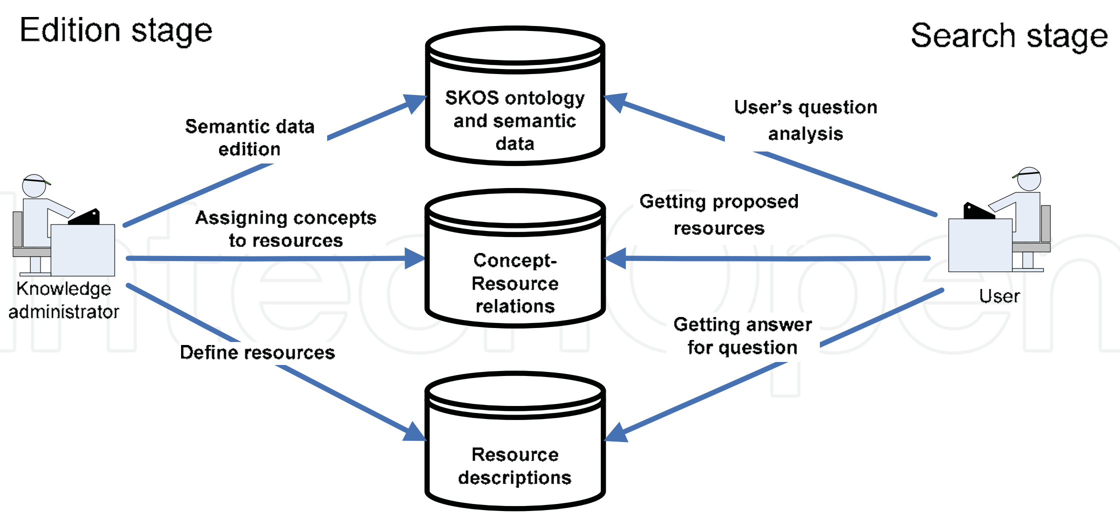

Fig. 2. The use of concepts defined in accordance with the SKOS ontology in the search process. [source: own].

The SKOS ontology has been supplemented by additional structures – sets of concepts which are directly connected to searched resources. The aim of the sets is to model a part of the user's question. The more sets are found in the user's question, the greater significance of the searched resource is. This way it is possible to model connections between concepts, based on the knowledge of a particular domain.
The last stage of the sentence analysis is the use of information about the words location with respect to one another in the user's sentence. The words which are closer to one another and point at the same resource simultaneously raise the priority of the found resource. This results from the prerequisite that, usually, the words which determine the same object are located close to one another in the sentence.
Such analysis allows to present the found resources to the user, according to the assigned search ranking.
Figure 3 shows a sample SKOS concepts structure and its relation to resources that are to be searched for. Three issues (real life situations) have been defined: finding an ID, losing an ID and getting a new ID. Additionally, the following concepts have been defined: finding, loss, theft, getting and issuing. The related relations allow to "strengthen" certain relations other than broader and relatedPartOf. With such defined relationships it is possible to address the questions about "robbery", or about "finding", both using the word "ID card" or "proof of identification".
Building a net of concepts and assigning resources to the concepts allow to model the system answers to the user's questions. This way the data administrator, who defines the system answer by himself/herself, has a clear picture of the system behavior with respect to a given class of questions. Such a solution is more deterministic than full-text search tools which operate on the basis of statistical methods only.

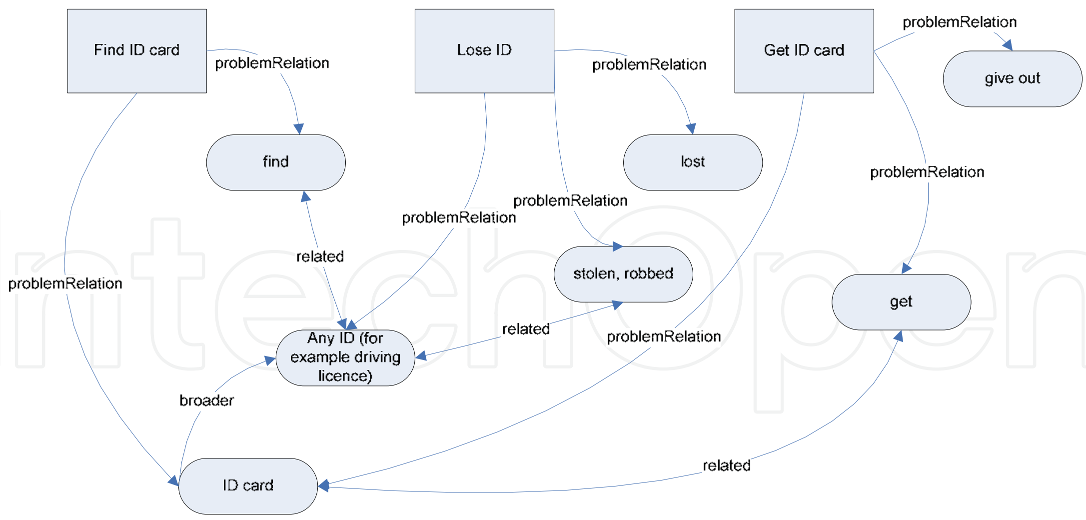

Fig. 3. Sample SKOS structure and its relation to the resources to be searched for [source: own].

Additionally, to improve the data administrator's operations in the system, the mechanisms were introduced which function in traditional search tools solutions, but at the edition stage of the ontology. Thus the possibility of automatic collection of concepts from the indexed elements (descriptions of life cases) was applied, and the process of assigning the concepts to life cases was automated. In order to perform this task, the algorithm was used to calculate normalized words priorities for documents (dt indicator) (Salton & Buckley, 1987). The algorithm allows to calculate the adequacy ranking of a given word for the indicated life case. Therefore the work with the tool can start from automatic indexing of life cases and then can proceed to successive introduction of revisions by means of successive introduction of relations between concepts, changing labels and their classification (pref, alt, hidden), etc.

2.4 Conclusions about search engine based on semantic networks

The presented solution is a proposal to solve a certain issue related to information search. It seems that the solution can improve the search in resources which are limited in terms of the number of indexed documents, and in the situation in which it is assumed that the users will ask "questions" to the search tool. The solution appears especially adequate in the case of the so called FAQ lists. They define ready answers to certain questions and, more importantly, the questions are usually relatively short. In such cases full-text search tools can have problems to properly index the contents.

On the basis of the conducted tests it seems that the efficiency of the search tool operations depends mainly on a well constructed ontology. Therefore the ontology is the key element which affects the functioning of the system. It is necessary to adopt a relevant methodology for building ontologies. The key issue in building an ontology is making it easy to manage in the future.

On the basis of the existing solution it is possible to introduce an extra feature – possibility to clarify the user's question. The proposed solution could be an engine which would control the conversation with the user in a specific manner.. In such a solution, in the first step the issue asked by the user would be found and then the engine should ask some questions (assigned to the chosen issue) to precise the question and give the most correct answer.

3. Natural language text generation

Recording knowledge and facts is related to the introduction of concepts, the features of these concepts, and relationships between concepts. Recording knowledge in the form of a natural-language sentence (descriptive text) contains the above mentioned concepts, dependencies and features. However, due to its nature, this way of recording limits the possibilities to process knowledge as well as to compare and connect similar concepts. Thus it is not possible to automatically combine knowledge from two different sources with the purpose to obtain some extra cross-sectional information based on two separate documents. Such cases refer particularly to the knowledge that resembles data structures where there is focus on certain dependencies between entities. An example is a description of a device and sub-assemblies the device consists of (catalogue of products, catalogue of sub-assemblies). The description will comprise not only typical information on a given sub-assembly but also the dependencies, e.g. which of other sub-assemblies is able to replace the given one, what other sub-assemblies it consists of, what material it is made of, etc. Similarly, scientific research results that contain parameters and their mutual dependencies can be described with the use of such a structure. This kind of solution enables to easily connect data from various sources and to find new dependencies.

Although this way of data description can be easily processed by a computer, it is less readable for the user. An ideal situation for the user would be the possibility to "question" the structural knowledge base with the use of a natural language and obtain answers in the form of grammatical sentences.

In this section the natural language generation will be described. In the first step storing knowledge as a semantic network will be described. Then we will show the state of the art in the natural language generation and specific issues connected with the Polish language. Next, a natural language generation engine based on semantic networks will be presented, which was built in the course of the Virtual Consultant for Public Services project.

3.1 Knowledge stored as semantic networks

As it has been already mentioned, semantic networks allow to describe concepts, their properties, classification of concepts and relationships between concepts. Semantic data can be used to describe documents (assigning tags), or they can be a source of knowledge by themselves. Ontologies which store knowledge have many properties and relationships defined between elements.

Sample ontologies for tagging texts, documents are:

- SKOS ontology[21] which allows to build vocabulary for a particular domain - concepts and relations between them. There are also special relationships between concepts: hierarchy, part-of relation, associations;
- OpenCYC[22] ontology which represents data within the CYC initiative. The main objective of the CYC Project is to collect concepts from real word and build relationships between them.

[21] http://www.w3.org/2004/02/skos/

Sample ontologies for knowledge management and storage are:

- FOAF ontology[23] which describes the person and the friends of that person, thereby creating a network of connected people;
- DBPedia[24] – ontology which provides data from Wikipedia in the structural way;
- Ontology for describing photos (Lafon & Bos, 2002);
- Ontology for describing spatial data[25].

Currently, the LikedData[26] initiative is promoting the idea to publish various types of already collected data as semantic data and combining them with each other. There are many tools available that help to publish data over HTTP as semantic data directly from existing relational databases. Thanks to such initiatives, the data published on the Internet will be readable not only for humans but also for different kinds of services, systems, applications. Computer systems will be able to use those data, combine them with each other and perform new functionalities.

Storing knowledge in such a way can be also used by programs that generate texts (readable for humans) from structural data.

3.2 State of the art in natural language generation

The natural language generation has been already described in many publications (Reiter & Dale, 1997), (Paris et al., 1991), (Cole et al., 1997). Some classes of software to generate text have been defined depending on algorithm complexity and quality results of a generated text. Some stages in the generation process were also defined.

The first stage is called text planning. At that stage it should be planned which part of knowledge should be described in a textual form.

In the next stage, the sentence content and order should be stipulated. The last stage is dedicated to generating sentences in a proper grammatical form. This process can be performed in different algorithm complexity, depending on language tools.

Natural language generation systems can be classified into (Cole et al., 1997):

- information systems which produce messages without infrastructure for text planning, sentence order and any language tools;
- systems which base on sentence templates. This solution depends on prepared templates which are filled with changeable elements;
- systems based on phrase templates. This solution depends on a part of sentence templates – phrases which are used recursively up to generate a meaningful sentence;
- systems based on sentences properties. In such solution, sentence templates with defined properties (question sentence, statement) are a starting point. Iterating through the successive stages, these templates are completed with additional details up to generate a meaningful sentence.

[22] http://www.cyc.com/cyc/opencyc/overview
[23] http://www.foaf-project.org/docs/specs
[24] http://dbpedia.org/About
[25] http://www.geonames.org/ontology/
[26] http://esw.w3.org

In the natural language generation it is important to use language tools specific for a particular language. They are especially important for inflective languages. It is important to use a proper grammatical form of the word: gender, tense, mode, plural/singular form.

3.3 Polish language specifics and language tools

Natural languages have different ways of building sentences. In English, the position of a word in the sentence is strictly determined. This facilitates the sentence analysis which, in turn, allows to precisely determine the meaning of the sentence. Polish is not a positional language. Verb, subject, attribute, etc. can occur in different positions in the sentence (Vetulani, 2004). However, Polish has fixed connections between parts of speech. These connections determine dependencies between particular parts of the sentence, i.e. the grammatical form of one part of the sentence enforces the grammatical form of the other part (Saloni & Świdziński, 1981). Unfortunately, these dependencies do not have strict character. They depend on the style of the sentences and their types too. For example, questions will have a different word order and different dependencies between forms compared to statements.

Polish language is also inflective. Depending on its gender, case, tense, a particular word is in a different form. Differences in grammatical forms are not manifested only by the endings of words. In comparison with other languages (for example English), there are many more irregular forms in Polish.

Some conclusions:

- Using the rules for sentence building can be very complex, especially for Polish. Formal description of the Polish language is carried out in IPI PAN. The formal description of the Polish language is defined in Gramatyka Świdzińskiego (Świdziński, 1992). There is also an implementation of that formalism, but it is on the experimental stage;
- It is important to use tools for getting a word in its primary form – lemma generation tool – when analyzing a piece of text;
- During the sentence generation it is necessary to use words in appropriate forms (correct case, gender, tense etc.). That is why the tool for generating words in their correct forms is needed.

When developing a natural language generation engine, the UTR tool was chosen for generating lemma and correct word forms. The author of this tool is Jan Daciuk. UTR uses a dictionary which contains words, their forms and form tags. Very good compression and easy browsing through words was achieved thanks to the finite-state automata algorithm. Technical details of the UTR tool have been widely described in the doctoral dissertation of Jan Daciuk (Daciuk, 1999).

3.4 Generator implementation

The described natural language generation engine has been developed as a system which depends on templates. The knowledge base for the generator is the data stored as a semantic network. The engine describes a concept stored in semantic data with the use of its properties and relationships between other concepts. The semantic network naturally provides the text planning stage. Sentence templates are connected to properties defined in the ontology. The lemma tool and the tool for generating forms provide proper forms of the generated text in template gaps.

```
<Flu> <has symptom> <Headache>
<Flu> <has symptom> <Fever>

<Flu> <treatment> <Aspirin>

<Aspirin> <contraindication> <Blood coagulability>
<Aspirin> <indication> <Flu>
```

Relationships between concepts

```
<Aspirin> <description> „Popular medicine. It is antiphlogistic and antithermic. Used in flu and other sicknesses.
```

Textual description

Fig. 4. Semantic data examples [source: own].

Browsing the knowledge base is primarily reviewing the different concepts, reading their properties and navigating between related concepts. The concepts describe some entities, so as a part of speech they are usually nouns. The properties define details about a concept or define relationships between concepts. That is why they are usually verbs or adjectives. Sample data are presented in Figure 4.

- for property `<has symptom>`
  `Objawy [[subject gen:sg:]] to [[object]]`
  (English: Symptoms …are…)
- for property `<treatement>`
  `Możliwe jest leczenie poprzez [[object acc:]]`
  (English: Treatement using…)
- for property `<contraindication>`
  `Nie może być [[$subject$ stosowany m:|stosowane n:|stosowana f:]] w przypadku [[object gen:]]`
  (English: It can't be used for….)
- for property `<indication>`
  `[[$subject$ Używany m: | Używane n:| Używana f:]] jest przy leczeniu [[object gen:]]`
  (English: It is used for…)

Fig. 5. Template examples [source: own].

The concept description consist of: textual description, properties (in fig: "has symptom", "treatment", "contraindication", "indication", "description") and relationships. Sentences are generated based on relationships between concepts and templates connected with properties (Figure 5). So the templates consist of the text which should be applied for a particular property and pointers which point where other elements from the RDF triple (subject and object) should be inserted.

In addition, the template contains information about the prescribed form of the expression to be inserted in the template. Information about the prescribed form can contain additional criteria: tense, gender, case, etc. It is not necessary to give each criterion. In such a case the

missing criteria will be preserved from the original word. For example, when for a particular field only tense is defined, the inserted words will preserve their gender and case.
To achieve some diversity in the generated text it is possible to define more than one template for the same property. In such a case one template will be chosen from the defined set of templates.
Some information may be stored as a text description. Property can point at a broader textual description. In such a situation it is presented as hyperlink.

3.5 Conclusions about natural language generation based on semantic data

The presented solution is an attempt to develop a service providing a universal method of searching and presenting the structural data sources. Not all types of information assets fit this model. Therefore the solution can find application first of all in such cases where the knowledge has an organized, structural character by nature. In the developed engine the most complex stage was choosing an appropriate form for the gaps in the template. Despite the use of markers that point at an appropriate form, there were some ambiguities and confusions. An idea to solve that issue is to use the Google browser to check which version of the phrase or part of sentence is more likely – which option has more results in the Google search engine. For future development it is also possible to build up a phase in which the knowledge is selected for generation and presentation. Currently, for the generation of the text "the nearest environment" of the concept is selected. One can imagine that further relationships are taken into account. It could be done by extending or changing algorithm used by reasoner engine.

4. Summary

The presented solutions combine the knowledge of the semantic networks and natural language processing. They verify the usefulness of the application of network-related issues in the semantic processing of a natural language. Both solutions use the powers of the semantic network in terms of modelling the relationships between concepts. The search engine, through the use of semantics, can better "understand" questions asked by the user. The impact is especially on the ability to define relationships (hierarchy, dependencies, relationships conclusion) between concepts. The mechanism of texts generation shows that semantic networks are a good way to store knowledge in a structural way with a flexible approach to modelling the relationships between properties and concepts. The possibility to describe properties (wide possibilities in metadata description) helps in developing an engine for generating text from the web of relationships. The final result – a generated text based on semantically stored knowledge makes information more readable for humans.
According to the presented solutions one can assume that using semantic networks can have good influence on other issues associated with the natural language. However, it is necessary to identify real needs in each case and define a proper place for using the semantic network in the developed solution.

5. References

Berners-Lee, T. (2001). *The Semantic Web*, Scientific American, (284(5)):34–43

Berners-Lee, T. (2006). *Relational Databases on the Semantic Web*, See: http://www.w3.org/DesignIssues/RDB-RDF

Brickley, D. (2004). *RDF Vocabulary Description Language 1.0: RDF Schema*, W3C

Bruijn, J. (2003). *Using ontologies*, DERI Technical Report, DERI-2003-10-29

Cole, R. & Mariani, J. & Uszkoreit, H. & Zaenen, A. & Zue, V. (1997). *Survey of the State of the Art in Human Language Technology*, Cambridge University Press, ISBN 0521592771

Daciuk, J. (1999). *Incremental Construction of Finite-State Automata and Transducers, and their Use in the Natural Language Processing*,, Politechnika Gdańska

Davies, J. & Fensel, D. & Harmelen F. (2003). *Towards the Semantic Web, Ontology-based Knowledge Management at Work*, John Wiley & Sons, LTD, ISBN 0470848677

Jupp, S. & Bechofer, S. & Stevens, R. (2008). *A Flexible API and Editor for SKOS*, The Univeristy of Manchester

Lafon, Y. & Bos, B. (2002). *Describing and retrieving photos using RDF and HTTP*, W3C

Manning, C.D. & Raghavan, P. & Schütze, H. (2007). *An Introduction to Information Retrieval*, Cambridge University Press, Draft,

McGuiness, D. & Harmelen, F. (2004). *OWL Web Ontology Language Overview*, W3C

Paris, C. & Swartout, W. & Mann, W. (1991). *Natural Language Generation in Artificial Intelligence and Computional Linguistic*, Springer, ISBN 0792390989

Przepiórkowski, A. (2005). *The Potential of The IPI PAN Corpus*, Institute of Computer Science, Polish Academy of Science, Warsaw

Reiter, E. & Dale, R. (1997). Building Applied Natural Language Generation Systems *Natural Language Engineering* Vol. 3, No. 01. pp. 57-87 Cambridge University Press

Saab, S. (2006). *The Semantic Web Revisited*, University of Koblenz-Landau

Saloni, Z. & Świdziński M. (1981). *Składnia współczesnego języka polskiego*, Wydawnictwo Uniwersytetu Warszawskiego, Warszawa

Salton, G. & Buckley, C. (1987). *Term weighting approaches in automatic text retrieval. Information Processing and Management* 32:431–443. Technical Report TR87-881, Department of Computer Science, Cornell University

Świdziński, M. (1992). *Gramatyka formalna języka polskiego*, Wydawnictwo Uniwersytetu Warszawskiego, Warszawa

Vetulani, Z. (2004). *Komunikacja człowieka z maszyną. Komputerowe modelowanie kompetencji językowych*, Akademicka Oficyna Wydawnicza Exit, ISBN 83-87674-66-4, Warszawa

W3C (2001). *Semantic Web Use Cases and Case Studies*, See: http://www.w3.org/2001/sw/sweo/public/UseCases

W3C (2007). *SKOS UseCase*, See: http://www.w3.org/TR/2007/WD-skos-ucr-20070516/

Zaihrayeu, I. & Sun, L. & Giunchiglia, F. & Pan, W. &Ju, Q. & Chi, M. & Huang, X. (2007). *From Web Directories to Ontologies: Natural Language Processing Challenges*, University of Trento - Italy - UNITN-Eprints

Engineering the Computer Science and IT
Edited by Safeeullah Soomro

ISBN 978-953-307-012-4
Hard cover, 506 pages
Publisher InTech
Published online 01, October, 2009
Published in print edition October, 2009

It has been many decades, since Computer Science has been able to achieve tremendous recognition and has been applied in various fields, mainly computer programming and software engineering. Many efforts have been taken to improve knowledge of researchers, educationists and others in the field of computer science and engineering. This book provides a further insight in this direction. It provides innovative ideas in the field of computer science and engineering with a view to face new challenges of the current and future centuries. This book comprises of 25 chapters focusing on the basic and applied research in the field of computer science and information technology. It increases knowledge in the topics such as web programming, logic programming, software debugging, real-time systems, statistical modeling, networking, program analysis, mathematical models and natural language processing.