A survey and classification of semantic search approaches

Christoph Mangold

Universität Stuttgart, IPVS,
Universitätsstr. 38, 70569 Stuttgart, Germany
E-mail: mangold@ipvs.uni-stuttgart.de

Abstract: A broad range of approaches to semantic document retrieval has been developed in the context of the Semantic Web. This survey builds bridges among them. We introduce a classification scheme for semantic search engines and clarify terminology. We present an overview of ten selected approaches and compare them by means of our classification criteria. Based on this comparison, we identify not only common concepts and outstanding features, but also open issues. Finally, we give directions for future application development and research.

Keywords: semantic search; Semantic Web; search engine; information retrieval; survey.

Reference to this paper should be made as follows: Mangold, C. (2007) 'A survey and classification of semantic search approaches', *Int. J. Metadata, Semantics and Ontology*, Vol. 2, No. 1, pp.23–34.

Biographical notes: Christoph Mangold is a PhD student at Universität Stuttgart. His PhD topic is about context aware document search in enterprise intranets.

1 Introduction

The Semantic Web provides a number of technologies to improve human and computer collaboration on the internet (Berners-Lee et al., 2001). One important issue centres on the management of documents and, particularly, the semantically supported document retrieval called semantic search.

Recently, a number of semantic search approaches have been published. Their application area and their realisation are diverse. However, they are based on a common set of ideas. With this survey, we identify and interrelate these ideas. We introduce a categorisation scheme to compare semantic search approaches and, thereby, establish a common vocabulary. We present ten selected approaches and compare them by means of our categorisation scheme.

We expect that this work will be useful for anyone who wishes to get an overview of current approaches to semantic search. This includes people from application development as well as from the research community. We consider it a first step to build a common understanding of ideas and approaches. We are convinced that there is a need for a survey of this area since, on the one hand, a considerable amount of research has been accomplished and, on the other hand, there are a number of open problems to be solved. With our work, we give an overview of current approaches and compare their underlying ideas. Based on this comparison, we identify a number of open issues for research and application development. We are aware that this presentation does not consider all approaches by any means, but we are confident of capturing the main ideas.

In this survey, we discuss approaches that exploit domain knowledge to process search requests. We present a broad range of domain knowledge utilisations that comprises ontology navigation, manual and automatic query modification and user context modelling.

The following example demonstrates a usage scenario of a semantic search engine. Consider a user who needs fundamental information on clusters. He inputs the keywords 'introduction' and 'cluster' to his semantic search engine, which knows, from earlier sessions, that the user belongs to the context of computer science. An ontology lookup tells the system that the term 'cluster' can have two meanings in the context of computer science. It prompts the user if he is looking for information on clusters related to computer networking or data analysis. According to the user's answer, the system finally retrieves introductory documents about clusters in data analysis. It not only returns documents that contain the term 'introduction', but also documents that contain 'overview' or 'fundamentals'.

1.1 What this paper is not about

In this survey, we consider approaches to retrieve text documents, only. We disregard search engines that work, e.g., on XML documents such as XSEarch by Cohen et al. (2003). We also disregard approaches that require the user to formulate queries in a formal language, such as RQL (Karvounarakis et al., 2002) or RDQL (Seaborne, 2004) for the RDF ontology description language. This also includes all solutions where the user is required to know more than mere keywords, such as the query-by-example approach by Banks et al. (2002) or the approach of Calvanese et al. (2004).

We consider search engines that focus on information retrieval from the (Semantic) web or special purpose information systems. We disregard approaches for peer-to-peer architectures, such as the approaches by Ding et al. (2004a), Calvanese et al. (2004) or the SWAP system by Ehrig et al. (2003).

We also do not discuss issues of non-semantic query expansion, such as those proposed e.g., by Mitra et al. (1998), or other issues of non-semantic information retrieval. For a survey in this area, see Mitra and Chaudhuri (2000).

For the sake of succinctness, we restrict ourselves to refer to only one publication for each approach or idea, although in many cases there are other publications of the same authors and projects. In these cases we tried to cite the most fundamental publication.

1.2 Definitions

Traditional document search mostly relies on the occurrence of words in documents. We define semantic search to be a document retrieval process that exploits domain knowledge.

Domain knowledge can be formalised by means of an ontology, which is often defined as an "explicit specification of a conceptualisation" (Gruber, 1993). For the rest of the paper we use the term *concept* to denote ontological classes/frames. The term *individual* represents instances/facts and *property* represents relationships/slots. We use *resource* to refer to an ontology element that may be a concept or an individual.

The goal of search engines is to maximise *precision* and *recall*, where:

$$\text{precision} = \frac{\text{Number\_of\_retrieved\_relevant\_documents}}{\text{Number\_of\_retrieved\_documents}}$$

$$\text{recall} = \frac{\text{Number\_of\_retrieved\_relevant\_documents}}{\text{Number\_of\_relevant\_documents}}.$$

Obviously, the maximum value for both parameters is 1.0. One way to increase precision and recall of a query is to exploit the semantic context of query terms. The most important concepts in this domain are the following. Let *a* and *b* be different terms.

- *a* and *b* are *synonyms* if they denote the same resource.
- *a* is a *homonym* if it denotes at least two different resources.
- *a* is a *hypernym* of *b* if the resource denoted by *a* is more general than the resource denoted by *b*. If *a* is a hypernym of *b*, then *b* is a *hyponym* of *a*.
- *a* is a *meronym* of *b* if the resource denoted by *a* is part of the resource denoted by *b*. If *a* is a meronym of *b*, then *b* is a *holonym* of *a*.

1.3 Overview

The rest of the paper is organised as follows. In Section 2, we introduce seven criteria that are useful to classify semantic search approaches. In Section 3, we summarise ten selected proposals for semantic search. In Section 4, we compare the approaches from Section 3 along the criteria we introduce in Section 2 and identify issues for further application development and research. Finally, Section 5 concludes the paper.

2 Classification categories

In this section, we present a categorisation scheme that we use to classify different approaches for semantic search along several dimensions. In particular, we introduce categories for the following criteria: Architecture, coupling, transparency, user context, query modification, ontology structure and ontology technology. The criteria we chose are not completely independent of each other. However, we feel that they capture important characteristics of semantic search engines.

We are aware that there are other criteria to classify semantic search engines. However, we do not discuss criteria that play (if at all) subordinate roles in the surveyed publications. The set of criteria we do not discuss is diverse and includes, e.g., performance/scalability, distribution, adaptability and the ranking of results. We would like to point out that we do not consider these issues unimportant – the contrary is true, see our list of open issues in Section 4. With our choice of categories we merely reflect what most authors regard as relevant.

In this section, we give an uncommitted overview only. In Section 4, we discuss and compare the criteria and identify semantically effective ideas.

2.1 Architecture

Just as for non-semantic search engines, there are two possible architectures:

- *Stand-alone search engine*. A stand-alone search engine consists of several parts. The crawler browses the document base. It stores document meta data in an index based on which the query engine evaluates query requests.
- *Meta search engine*. A meta search engine does not maintain an index of documents itself. It distributes queries to other subordinate search-engines and combines the results afterwards.

In our survey we found several semantic search engines incorporate standard search engines as part of their architecture. We consider these engines as meta engines only if the subordinate search engine is a standard search-engine that does not need to be customised or otherwise altered.

2.2 Coupling

One of the most obvious classification criteria is the coupling between documents and ontologies. There are two approaches:

- *Tight coupling*: In tightly coupled approaches the meta data of documents refer explicitly to concepts of a specific ontology or vice versa. Sometimes, documents are considered as individuals in an ontology. With this approach, homonymies can be resolved easily by choosing the appropriate concept from the ontology. However, this comes at the (potentially high) cost of semantic document annotation. The whole research area of 'information extraction' (e.g., Ciravegna et al., 2002) is dedicated to this problem. Tightly coupled approaches are applied not only to special purpose information systems that manage data of a limited organisation or domain but also to more general application areas such as the web.
- *Loose coupling*: We speak of loose coupling if documents are not committed to any available ontology. In the loosely coupled case, there is the problem of choosing an appropriate ontology for a given domain. Consequently, the semantic power of loosely coupled systems is limited, since e.g., homonymies can not be resolved that easily. However, in the World Wide Web (WWW) scenario where (at the time) only a very small fraction of documents is semantically annotated this offers a feasible approach. Loosely coupled systems can be implemented as meta search engines easily.

The coupling criterion is important since loose coupling restricts the search capabilities whereas tight coupling requires annotated documents. We further discuss the implication of coupling and architecture in Section 4.

2.3 Transparency

Regarding the user interaction with semantic system-features, we found the following transparency types in our survey:

- *Transparent*: The semantic capabilities of the system are invisible to the user; the system appears to be an 'ordinary' search engine. Transparent systems have no means to request additional information from the user, e.g., to clarify homonyms.
- *Interactive*: Interactive systems may ask the user for clarification or recommend changes to the query. These systems are sometimes called 'recommendation systems'.
- *Hybrid*: Hybrid systems combine interactive and transparent behaviour. In the standard case they act as transparent systems. They require user interactions for very specific tasks, only.

Transparency is an ambivalent feature. On the one hand, the user is spared with lengthy system dialogs. This makes a system easy to use. On the other hand, the user can not influence the semantic decisions of the system, which potentially leads to low precision. We included transparency in our categorisation, since it plays a major role in user acceptance.

2.4 User context

The usefulness of documents always relates to the user context. Many semantic search engines apply the user context to better meet the user's information need. We distinguish the following classes:

- *Learning*: User context is extracted from user interaction dynamically. Based on the user's query and query-refinement history the system guesses about desired results. If query terms always belong to the same ontological context, the system can presume the resolution of homonymies.
- *Hard-coded*: In the hard-coded approach, queries are categorised in so-called *question-categories* that specify the user's information need. The system provides a fixed number of question-categories that are exploited during query evaluation. Typical categories can define the kind of information need such as "location of …", or "general resources for …" or the context of the information need such as 'Jazz'. In the latter case, if the user searches for 'Miles' the system will return no documents about the Miles College or Air Miles but documents on Miles Davis, only.

  The assignment of a user's query to a question-category can be done explicitly by the user or implicitly based on the user's user-group, or by analysing the query. Usually, in ontology-based solutions information-need classes correspond to certain ontology structures such as entry points and property types.

The User-context provides important information about the user's information-need. This information is employed for query modification.

2.5 Query modification

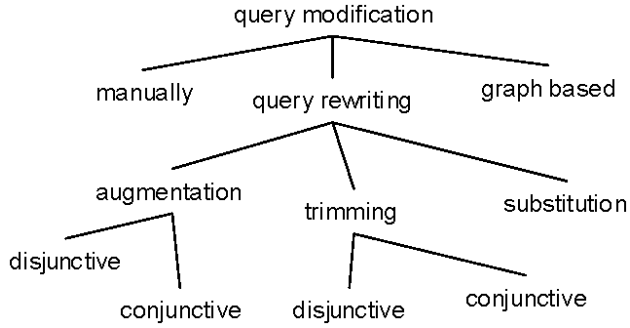
The semantic modification of user queries is a well-known technique from information retrieval, see e.g., Mitra et al. (1998). In the area of semantic search it often exploits information from ontologies. It plays a central role in many semantic search engines. Different techniques have been developed to increase both, recall and precision of a query. The increase of precision is often called *query disambiguation*.

In the presence of ontologies, it is relatively easy to increase the recall of a query. In this case, the ontology supports the search engine with terms that are more general than, or otherwise related to, the query-terms.

E.g., if we replace a search term with its hypernym, not only the documents that are committed to the search term itself but also the documents that are committed to other hyponyms of the search term's hypernym are retrieved. In contrast, it is a hard task to improve the precision of a query. This involves, e.g., the resolution of homonymies and of the hyponymy problem, i.e., when a user searches for documents related to an ontological concept, the query precision increases when using a specific sub-concept instead of the concept itself.

We give an overview of query-modification techniques in Figure 1. Roughly, they fall into the three categories of manual query modification, modification by query rewriting and graph-based query modification. In the following we detail each of these categories.

Figure 1 The variants of query modification



- *Manually*: The simplest way to modify a query leaves the modification to the user. When the user enters a query, the system returns not only documents, but also an appropriate part of an ontology. The user navigates the ontology and reformulates his query, i.e., adds or removes query terms.
- *Query rewriting*: Query rewriting is driven by the idea that a query can be optimised by the system. We observe three different ways, augmentation, trimming and term substitution.

  In the case of *augmentation*, the query is enhanced with terms that are derived from the ontological context of the original query terms, e.g., the query for 'Einstein' could be enhanced with 'theory of relativity'. Depending on the ontology structure (see next subsection) different semantics can be exploited. The *trimming* of a query removes query-terms and has the opposite effect of augmentation. Query trimming can be realised by comparing the results of a trimmed query with the results of the original query. E.g., when the query for 'Einstein' and 'theory of relativity' and 'black hole' yields no results, the search engine may find that omitting the term 'Einstein' yields a reasonable result set and may suggest the trimmed query to the user.

  Augmentation and trimming exploit that a query consisting of a *Conjunction* (AND) of terms becomes more specific with each additional term, where a query composed of a *Disjunction* (OR) becomes more general. In other words, related to the user's information need, long conjunctive queries yield high precision, where long disjunctive queries lead to high recall.

  Hence, both techniques, disjunctive query augmentation and conjunctive query trimming increase the recall where conjunctive query augmentation and disjunctive query trimming increase the precision of a query. Many systems provide either conjunctive or disjunctive queries, i.e., the search terms are implicitly and invariably connected with either AND or OR. Consequently, the idea of disjunctive augmentation or trimming is not applicable to a system that only provides conjunctive queries.

  We speak of *substitution* when search terms are replaced with ontologically related terms. In general, terms are substituted with synonyms, hypernyms or hyponyms from the ontology to increase recall or precision, respectively. We treat substitution separately since it differs from augmentation and trimming in the following respect: compared to the result set of the original query, disjunctive augmentation and conjunctive trimming yield a super-set of results where disjunctive trimming and conjunctive augmentation produce sub-sets. In contrast, substitution may yield a result-set that only partially overlaps the original result set. In a scenario where the user submits several queries iteratively to satisfy a certain information need, substitution appears to be the more effective technique to guide the user according to his evolving domain knowledge.
- *Graph-based*: The third technique to optimise user queries requires tight coupling between the document base and the ontology. It perceives both, ontological concepts and documents as the nodes of a graph. Query terms are used to find relevant nodes in the graph. From these nodes, an algorithm traverses the graph to determine semantically related documents. This task can be achieved, e.g., by a spreading-activation algorithm. Graph-based query-modification differs from query rewriting in that it does not construct a new query that is subsequently processed by a search engine, but it directly returns relevant documents. Furthermore, it considers the query as a whole instead of decomposing it into terms.

2.6 Ontology structure

Ontology-based semantic search engines rely on certain ontology structures. Ontologies are usually built from concepts, properties, constraints and possibly axioms. We observe that semantic search exploits properties only and distinguish the following cases:

- *Anonymous properties*: In the case of anonymous properties, the system disregards the name and the semantics of the property. The interrelation between two concepts indicates that they share the same context, only.
- *Standard properties*: We only found a small set of 'common sense' properties in our survey. The properties are synonym_of, hypernym_of, meronym_of, instance_of and negation_of. The homonym_of property does not have to be modelled explicitly since it is equivalent to term equality. The usage of standard properties enhances semantic search capabilities. However, it also introduces dependencies on ontological structures. For an overview of how standard properties can be exploited for semantic search see, e.g., Bates (1990).
- *Domain specific properties*: Besides standard properties, a system can exploit domain specific properties, as e.g., 'camera type' in a photograph retrieval system.

In fact, systems may choose any combination of the three types, e.g., to use only a subset of the standard properties and treat the rest as anonymous properties. Ontology structure is an important criterion since it characterises the flexibility of the search engines concerning the reuse of ontologies.

2.7 Ontology technology

To express ontologies, the ontology description language is always of interest. Whereas the ontology structure determines the *semantic* reusability of ontologies, the ontology technology focuses on *technological* reusability and interoperability. Besides, several solutions that use proprietary languages the most widespread technologies contain *F*-Logic (Kifer et al., 1995), RDF (Manola and McBride, 2004), DAML(+OIL) (Horrocks, 2002) and OWL (McGuiness and van Harmelen, 2004). Some approaches also use the lexical database from WordNet (Fellbaum, 1998).

3 Overview of selected approaches

For this survey, we studied 22 different approaches and projects about semantic document retrieval. It would take too much space to present them all. We selected ten of them according to the following criteria. We favoured complete, detailed and traceable descriptions that focus on the issue of semantic search. We paid attention to present at least one approach for each of the aforementioned characteristics. Furthermore, instead of discussing the varieties and the evolution of selected ideas we prefer to present a wide spectrum of approaches that shows the diversity of ideas.

Particularly, we discuss the following solutions: Simple HTML Ontology Extensions (SHOE) (Heflin and Hendler, 2000), Inquirus2 (Glover et al., 2001), TAP (Guha et al., 2003), Hybrid Spreading Activation (Rocha et al., 2004), Intelligent Semantic Web Retrieval Agent (ISRA) (Burton-Jones et al., 2003), Librarian Agent (Stojanovic, 2003), Semantic Content Organisation and Retrieval Engine (SCORE) (Sheth et al., 2002), TRUST (Amaral et al., 2004), a system for audio data retrieval (Khan et al., 2004) and Ontogator (Hyvönen et al., 2003).

The approaches we surveyed but do not discuss here are WebSCSA (Crestani and Lee, 1999), QuizRDF (Davies and Weeks, 2004), OntoIR (García and Sicilia, 2003), Swoogle (Ding et al., 2004b), OWLIR (Shah et al., 2002), Swangler (Finin et al., 2005), a dynamic reasoning engine by Jelmini and Marchand-Maillet (2004), SemanticMiner (Moench et al., 2004), Ontoseek (Guarino et al., 1999), a system for semantic search in annotated Chinese poetry by Soo et al. (2004), Freedom (Semagix, 2005) (formerly: Taalee) and KAON (Maedche et al., 2003).

In this section, we present only the approaches. In Section 4 we compare them and discuss the semantically most effective solution. In presenting the approaches, it was not our goal to give a comprehensive introduction to each approach but to provide a succinct overview of characteristic ideas. Detailed information can be found in the cited literature. In general, we had to face the problem that some publications describe their approaches from a very abstract viewpoint. In these cases, we relied on the given information without knowing the details, e.g., the exact algorithm that lies beneath. We are aware that we did not look at all approaches in the quickly developing field of Semantic Web technologies but we hope to cover the basic ideas.

3.1 Simple HTML Ontology Extensions (SHOE)

An early approach to realise document retrieval in the Semantic Web has been presented in the scope of the SHOE project by Heflin and Hendler (2000). The SHOE approach requires a domain-ontology where document types correspond to ontology concepts. E.g., for a university homepage the ontology may contain concepts like 'faculty homepage', 'project homepage', or 'graduate student homepage'. Furthermore, the ontology contains properties of concepts that denote, for e.g., the name of the student or that he may 'work for' a specific project. Individual web pages commit themselves to ontological concepts and property types by means of the SHOE markup-language that is invisible to browsers but visible to semantic-aware search engines. Hence, web pages are individuals. E.g., the homepage of project *p* is an instance of concept 'project homepage' and there is a property that connects the page with the homepage of a certain student *s*.

In a semantic-search scenario, the user chooses one concept from the ontology. The system responds with a set of properties that are applicable to the selected concept. Subsequently, the user specifies values for properties he is interested in from which the system generates a conjunctive query and evaluates it on the document base. E.g., the user chooses the concept 'graduate student homepage' and specifies the value of the name-property as 'Peter'.

Then, the system returns graduate student homepages that belong to students with name 'Peter'.

SHOE requires tight coupling between concepts and web pages and has a stand-alone architecture. The system has no notion of user context. Query modification involves manual navigation of the concept hierarchy and concept-properties.

Similarly to SHOE, the OntoIR system by García and Sicilia (2003) belongs to the class of navigational approaches. It improves the SHOE system and mainly focuses on user interface issues. It is capable of exploiting arbitrary RDF and DAML + OIL ontologies and relies on their concept hierarchies, mainly.

The QuizRDF System by Davies and Weeks (2004) also follows the main ideas of the SHOE system. It enhances a full text index with ontological information. Hence, QuizRDF can do a combination of both, ontological navigation and full text search.

In terms of our classification criteria, OntoIR and QuizRDF only slightly differ from SHOE, so we do not discuss them in detail.

3.2 Inquirus2

The Inquirus2 approach by Glover et al. (2001) descends from the family of web search-engines. It is implemented as a transparent meta search-engine that uses question-category based query-modification to better meet the users' information-need.

When Inquirus2 receives a query and a question-category, it enhances the query with additional search terms, selects appropriate search engines and submits the enhanced query. Subsequently, it combines and ranks the results and finally presents them to the user. In this process, the question category influences not only query enhancement but also selection of search engines and combination and ranking of results. For example, if the user wants to retrieve general resources of a subject, the query is enhanced with 'what is' and 'links resources'.

The Inquirus2 system does not employ ontologies but relies on hand-coded rules for each question-category. It offers no possibility to exploit ontological domain information for query modification.

3.3 TAP

The Semantic Web application framework TAP presented by Guha et al. (2003) combines traditional information retrieval with semantic search. Here, the Semantic Web is considered an RDF-ontology that is separate from ordinary web pages. The semantic-search facility is an independent add-on to ordinary text search. Consequently, query-results consist of two parts. On the one hand, there is a list of documents retrieved by means of ordinary text search. On the other hand, the result contains a subset of the ontology that is relevant for the given query, i.e., a set of RDF triples. (For another approach that focuses on the retrieval of ontologies only, refer to the Swoogle system by Ding et al. (2004b), which we do not discuss in this survey).

The problem of query disambiguation in the ontological part is addressed in three different ways: first, by measuring the distances between query terms based on the distance in the RDF graph, secondly by exploiting the user context and thirdly by measuring the popularity of the term in the document base. The TAP framework realises two different ideas to incorporate user context. First, it exploits the user's query history. Secondly, it specifies user contexts explicitly by tagging parts of the ontology. If a user submits a query, the system calculates the query-term context from a designated part of the ontology, only. If this leads to unsatisfactory recall, the system considers the entire ontology.

Similar to Inquirus2, query disambiguation for the document part uses question-categories. However, we could not find the details about how they are realised technically.

Since the Semantic Web and the document base are independent, we classify this approach to be loosely coupled. We file it as meta search-engine, because it relies on a standard IR search engine.

3.4 Hybrid spreading activation

The Hybrid spreading activation approach by Rocha et al. (2004) requires tight coupling between the document base and the ontology. Web pages play the role of individuals in an ontology that complies with a domain-specific schema. The ontology is a graph where concepts and properties are nodes and edges, respectively. Query execution roughly consists of two steps: First, for a given query a standard text search engine determines a set of nodes that are matched by the given query terms. Subsequently, these nodes are used as the start nodes of a spreading activation algorithm. Consequently, we classify the query modification as 'graph-based'. In general, spreading activation algorithms run on graphs that represent concepts and their mutual associations. The 'activation' of a node represents its importance. It depends on the node's start activation and the sum of the activation of associated nodes multiplied by the strength of association. In the approach by Rocha et al. (2004), documents with highest activation are ranked highest in the result set.

The approach is a stand-alone architecture. The semantic capabilities are transparent to the user. The system does not exploit user contexts.

Crestani and Lee (1999) proposed a very similar approach that uses spreading activation. We do not discuss it further in this survey, since it is a predecessor to the hybrid spreading activation approach by Rocha et al. (2004), which also comprises its main ideas. It does not support keyword search but retrieves documents related to a given set of example documents.

3.5 Intelligent Semantic Web Retrieval Agent (ISRA)

The ISRA proposed by Burton-Jones et al. (2003) corresponds to the pattern of meta search-engines. The system is loosely coupled, i.e., the ontology is independent of the document-base. The approach focuses on query modification that exploits information from WordNet and from DAML concept-hierarchies.

For each query, the ISRA system generates a small semantic network to capture the meaning of the query. The semantic network does not only contain the query terms and their synonyms and hypernyms, but also allows for negated terms. It enables the system to guess the correct term senses and resolve inconsistencies. From the semantic network, the system extracts an enhanced boolean query that is sent to subordinate search engines. Although the query modification of ISRA is based on a graph structure, it belongs to the category of query-rewriting since it produces another query and does not directly return documents.

Regarding the transparency criterion, we classify the approach as hybrid. The user's feedback is required in the presence of irresolvable homonymies, only. The system has no notion of user contexts.

3.6 Librarian agent

The Librarian agent system by Stojanovic (2003) behaves like a human librarian. Users refine their information needs in an interactive process. The system is a stand-alone search-engine. It uses tight coupling between documents and ontology.

Query processing involves three information sources:

- the ontology is used to determine the clarity or unambiguousness of a query
- the user's past queries help to guess the correct meaning of query terms
- the document-base is analysed to predict the result-set size of augmented or trimmed queries.

The approach to exploit the document-base is unique in our survey. It originates from the notion that the worthiness of result-sets corresponds to their size. E.g., when the user enters 'Einstein' AND 'relativity theory' the system returns 20 documents and the hint that the trimmed queries 'Einstein', OR 'relativity theory' yield 189 or 211 documents, respectively, where the augmented query 'Einstein' AND 'relativity theory' AND 'special' yields 12 documents, only. This way, the system guides the user to refine his query in an iterative and interactive process.

The Librarian agent supports conjunctive queries, only. It does not rely on certain ontology structures, i.e., all ontology properties are treated the same.

3.7 Semantic Content Organisation and Retrieval Engine (SCORE)

The SCORE system by Sheth et al. (2002) incorporates basic ideas in the area of document meta-data management and Semantic Web that have been licensed to a company called Taalee that has become part of Semagix Ltd. Today, Semagix (2005) offers a product called FREEDOM that we do not discuss in detail.

SCORE – short for Semantic Content Organisation and Retrieval Engine – embraces a broad spectrum of semantic technologies, which includes semantic meta-data extraction from text, document classification and semantic search. It is targeted towards enterprise intranets. The system features a stand-alone architecture.

Its semantic search engine requires a tight coupling between documents and ontology. However, the system can produce this coupling itself by means of the aforementioned meta-data extraction capabilities. As Sheth et al. (2002) describe, there is no automatic query modification. To exploit the semantic resources of SCORE, the user needs basic knowledge about the ontology structure.

3.8 TRUST

The TRUST semantic search engine by Amaral et al. (2004) realises semantic document retrieval in a multi-lingual question answering system. The system allows tight and loose coupling between documents and ontology. However, semantic search mechanisms are more elaborate in the tightly coupled case. The TRUST engine is the only approach in this survey that has a hybrid architecture. In the loosely coupled case it plays the role of a meta-search engine. However, for tightly coupled documents it maintains its own index structures.

Query modification in the TRUST approach bases on an ontological concept hierarchy, linguistic information and predefined question-categories. However, Amaral et al. (2004) are not specific about the query modification algorithm, which prohibits a classification. In addition, we could not determine if the system is interactive or transparent.

3.9 Audio data retrieval

The audio retrieval proposed by Khan et al. (2004) is part of a special-purpose information system. It retrieves news items from a collection that is fed by broadcast audio-streams. The audio meta data are extracted by speech recognition and from plain-text content-descriptions that are supplied by the broadcast stations.

The approach contains disjunctive query augmentation and term substitution based on a domain ontology. The ontology consists of concepts, individuals and their

synonyms, hypernyms and meronyms. The 'upper part' of the ontology is designed manually, while the lower-level concepts are extracted from the Yahoo hierarchy (Labrou and Finin, 1999). There is no notion of user context.

3.10 Ontogator

The Ontogator system by Hyvönen et al. (2003) is part of an image management and retrieval system. Images are annotated with terms from n RDF ontology. Ontogator does not offer automatic query modification. However, an interactive recommendation system allows the user to browse images based on ontological properties. Ontogator comprises an interesting feature to exploit user-contexts. It introduces views to the ontology that rely on different concept hierarchies, called 'facets'. Each view represents a specific information-need.

The Ontogator recommendation-system relies on a domain-specific ontology structure. However, Hyvönen et al. (2003) also propose a mapping-approach to deal with ontologies that have been designed for a different purpose, i.e., ontologies that miss the required structure or that are too detailed.

4 Evaluation

In this section, we compare the systems presented in Section 3 by means of the classification criteria introduced in Section 2. Subsequently, we discuss issues that are open to further research and application development. To the best of our knowledge, our conclusions are also valid against the background of the other 11 systems that we surveyed but did not discuss in detail.

4.1 Comparison

In Table 1(a) and (b), we give an overview of the results of our survey. In cases where we could not gather unambiguous information for certain criteria, we denote 'unclear' in the respective table entry. If a system combines functionality from different classes of a category, it is called 'hybrid'. In the last column we denote an imaginary 'semantically most effective' system. The entries of the semantically most effective system denote the most powerful and comprehensive idea for each entry or 'open' if the quantitative comparison between competing ideas is an open research issue. We consider a concept most powerful if it exploits all available information.

In the first row we denote the *focus* of each approach. This is not part of our classification since it is no concise criterion to distinguish systems. Nonetheless, we include it in Table 1 to recall the background of the respective system. We distinguish between systems that are targeted towards the WWW in Table 1(a) and search engines that are part of self-contained information systems in Table 1(b).

Comparing the *architecture* and the *coupling*, we observe that tightly coupled stand-alone systems prevail in the area of self-contained information systems. Approaches for the WWW include both, loosely-coupled meta search-engines and tightly-coupled stand-alone solutions. Furthermore, it is interesting to observe that there is no system that combines tight coupling with a meta-search architecture. Such a system would have to be a meta search-engine that incorporates subordinate *semantic* engines. This approach would require a concept for the mapping between the ontologies of subordinate engines and the meta-engine ontology.

We are not aware of any comparison between stand-alone and meta architectures concerning semantic effectiveness, so we declare it as 'open'. As for the coupling criterion, on the one hand tight coupling enables more powerful semantic retrieval methods. On the other hand, it requires semantic document annotation which is a big issue. Hybrid coupling subsumes tight and loose coupling. Hence, in heterogeneous environments such as the internet or the intranet where annotated documents coexist with un-annotated documents a hybrid solution will be most effective.

The *transparency* criterion denotes if the semantic capabilities of the system are transparent to the user. Here, we found a broad spectrum of approaches. In our view, the semantically most effective solution provides both, transparency for inexperienced users and interactive behaviour to experts. We could not find any results on user acceptance of interactive semantic search features. Again, we think that a hybrid system is most effective.

The *user context* can help to increase the precision and recall of a query. We found relatively few details on both approaches, hard-coded and learning. In particular, it seems to be an open issue how to annotate user contexts to ontologies with unknown structure. For instance, the Ontogator system is aware of the complex structure of its underlying ontology, which also models user context. However, we found no concept that generalises this approach to ontologies with little or unknown structure.

In our view (and as Guha et al. (2003) argue) the approaches of hard-coded and learning user-context do not interfere. They can coexist in a system and contribute two different important facets of user context. Hence, we classify the combination of both ideas as semantically most effective.

As pointed out above, *query modification* plays a central role in semantic search engines. Accordingly, we found a wide variety of approaches. However, query augmentation is more popular than query trimming. Only the approach by Stojanovic (2003) uses conjunctive trimming and none of the surveyed systems uses disjunctive trimming. Comparing graph-based query modification with query rewriting we learned that, in general, query rewriting offers more parameters to

optimise precision and recall of a query, where graph-based modification requires less semantic structure in the ontology. However, graph-based modification approaches are more likely to raise performance issues. We are not aware of any results that compare the effectiveness of different query modification approaches so we denote 'open' for the most effective system. As for the query rewriting techniques, from a conceptual viewpoint the different approaches do not interfere with each other and, hence, could be combined. However, we see no evidence that the more is automatically the better.

Table 1(a) Comparison of semantic search approaches. Entries in italics indicate that they are semantically most effective (see Table 1(b))

Prototype/project	*SHOE*	*Inquirus2*	*TAP*	*Hybrid spreading activation*	*ISRA*	*Librarian agent*
By	*Heflin and Hendler* (*2000*)	*Glover et al.* (*2001*)	*Guha et al.* (*2003*)	*Rocha et al.* (*2004*)	*Burton-Jones et al.* (*2003*)	*Stojanovic* (*2003*)
Focus	WWW	WWW	WWW	WWW	WWW	WWW
Architecture	Stand-alone	Meta	Meta	Stand-alone	Meta	Stand-alone
Coupling	Tight	–	Loose	Tight	Loose	Tight
Transparency	Interactive	Transparent	*Hybrid*	Transparent	*Hybrid*	Interactive
User context	None	Hard-coded	*Hard-coded and learning*	None	None	Learning
Query modification	Manually	Conj. augm.	Ontology-part: graph-based Document-part: unclear	Graph-based	All sorts of query rewriting	Conj. trimming Conj. augm. Substitution
Ontology structure	Hypernym Anonymous	–	Anonymous	Domain-specific	Hypernym Synonym Negation	Anonymous
Ontology technology	Proprietary	–	*RDF*	Unclear	*DAML concepts + Word net*	Proprietary

Table 1(b) Comparison of semantic search approaches. Entries in italics indicate that they are semantically most effective. For referential usage the last column denotes the semantically most effective idea

Prototype/project	*SCORE*	*TRUST*	*Audio data retrieval*	*Ontogator*	*Semantically most effective*
By	*Sheth et al.* (*2002*)	*Amaral et al.* (*2004*)	*Khan et al.* (*2004*)	*Hyvönen et al.* (*2003*)	
Focus	Information system	Information system	IS for audio data retrieval	IS for image retrieval	–
Architecture	Stand-alone	Hybrid	Stand-alone	Stand-alone	Open
Coupling	Tight	*Hybrid*	Tight	Tight	Hybrid
Transparency	Interactive	Unclear	Transparent	Interactive	Hybrid
User context	Unclear	Hard-coded	None	Hard-coded	Hard-coded and learning
Query modification	Manually	Conj. augm.	Disj. augm. Substitution	Substitution	Open
Ontology structure	Hypernym Domain Specific	Hypernym Synonym Rest unclear	Hypernym Synonym Meronym Instance_of	Hypernym Meronym Domain-specific	Facultative sum of all
Ontology technology	Unclear	Proprietary	Proprietary	Proprietary	Standard(s)

The surveyed approaches also show a large heterogeneity concerning *ontology structure*. However, we observed that there exists a common subset of properties that are used by approaches that do not treat all properties anonymously. This subset consists of hypernyms and synonyms.

The more ontology structure is available to the system the better it can support semantic search. However,

if the system requires only little ontology structure it is more flexible regarding ontology evolution, ontology integration and ontology replacement. We envision that the semantically most effective system can make use of the entire set of standard properties as explained in Section 2, but does not require them, i.e., it adapts its semantic search behaviour dynamically according to the given ontology.

The *ontology technology* contributes to ontology exchangeability on the syntactic level. Here, standards like RDF or OWL are the best choice not only concerning exchangeability but also in terms of tool support like ontology editors and reasoners.

4.2 Areas of further application development and research

In this subsection, we summarise some open research issues. We are aware that these topics are by no means exhaustive. On the contrary, we are convinced that many classification criteria themselves need further detailed study, such as the above mentioned mapping of user context to arbitrary ontologies. However, the following list reflects what we expect to be important in future research and development of semantic search engines.

- *Analysis of query modification*. To the best of our knowledge, there is no quantitative comparison of query modification techniques. A framework to evaluate query results such as provided by TREC[1] for standard information retrieval would be a first step in this direction.
- *Meta semantic-search*. We surveyed several meta search engines that modify user queries and propagate them to subordinate standard search engines. However, we found no concept to incorporate subordinate semantic search engines. With the success of meta crawlers in standard information retrieval and the growing number of semantic search engines we think that this approach deserves investigation.
- *Analysis of user acceptance*. The surveyed systems show a broad spectrum of transparency. The more interactive it is, the more powerful a system may be. Yet, it is unclear how much semantic interaction a user is willing to bear to improve his search results. In other words, we need to analyse what sort of interaction pays off for the user. To solve this issue we expect the research community to cooperate closely with application development.
- *Adaptability*. Many systems require a certain ontology structure, i.e., they rely on custom-tailored ontologies. Other systems – classified as 'anonymous' – cope with arbitrary ontologies but provide weaker semantic capabilities. It is an open problem how systems may adapt themselves to *existing* ontologies, i.e., ontologies that have been designed with a different purpose. This is not only important concerning the reuse of ontologies but also as regards the interoperability between knowledge-based systems in general. We consider the system adaptability as an important step towards domain-independent semantic search engines.
- *Ranking*. To our surprise, we found only a few approaches that contain ontology-based document ranking (Khan et al., 2004; Rocha et al., 2004) so we do not discuss it in this survey. However, from standard information retrieval we learn that ranking is among the most important functional issues of search engines. Hence, we expect research in this area as well.
- *Integration with DMS/CMS*. All surveyed approaches either focused on the WWW or support a special-purpose information system. No approach integrates with standard Document or Content Management Systems (DMS/CMS) such as, e.g., IBM's Content Manager (Zhu et al., 2004). We feel that information stored in DMS/CMS provides a good basis for semantic search engines. We expect that maturating semantic search technology will integrate with off-the-shelf DMS/CMS, soon.
- *Performance/scalability*. We only found few work on the performance of systems. On the market, semantic search engines have to compete with standard search engines. They may introduce only little overhead compared to standard solutions. Consequently, they need efficient implementation regarding indexing time, index space and response time.

5 Conclusion

In this work, we introduced a classification scheme for semantic search engines. With regard to the classification scheme we explained common ideas, their advantages and drawbacks. We surveyed 22 systems, ten of which we presented and compared by means of our classification. We discussed which ideas are semantically most effective for each classification criterion. Furthermore, we identified research and application-development issues that are not addressed by current systems.

From this survey, we learn that there are a large number of promising approaches to semantic document retrieval. However, for the area to mature it takes two crucial requirements. On the one hand, the research community needs to fill a number of gaps, as discussed in Section 4. On the other hand, we need application developers to transfer and validate promising concepts. In our view, it requires the synergetic cooperation of both groups to bring semantic document retrieval to its full potential.

Acknowledgements

The author would like to thank Bernhard Mitschang, Holger Schwarz and Thomas Schwarz for commenting on drafts of this survey and fruitful discussions that helped to develop and consolidate the work during several

iterations. This research was supported by the German Research Foundation (DFG) within the scope of the Collaborative Research Centre (SFB) 467.

References

Amaral, C., Laurent, D., Martins, A., Mendes, A. and Pinto, C. (2004) 'Design and implementation of a semantic search engine for Portuguese', *Proceedings of 4th International Conference on Language Resources and Evaluation, LREC 2004*, Lisbon, Portugal, Vol. I, pp.247–250.

Banks, D., Cayzer, S., Dickinson, I. and Reynolds, D. (2002) 'The ePerson snippet manager: a semantic web application', *Technical Report HPL-2002-328*, HP Laboratories, Bristol, Vol. HPL-2002-328 20021122, 84 pages.

Bates, M.J. (1990) 'Where should the person stop and the information search interface start?', *Information Processing and Management*, Vol. 26, pp.575–591.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The semantic web', *Scientific American*, Vol. 2001, No. 5, May.

Burton-Jones, A., Storey, V.C., Sugumaran, V. and Purao, S. (2003) 'A heuristic-based methodology for semantic augmentation of user queries on the web', *Conceptual Modeling – ER 2003, 22nd International Conference on Conceptual Modeling*, Chicago, IL, USA, October 13–16, *Proceedings*, pp.476–489.

Calvanese, D., Giacomo, G.D., Lembo, D., Lenzerini, M. and Rosati, R. (2004) 'What to ask to a peer: ontology-based query reformulation', *Proc. the 9th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2004)*, AAAI Press, Whistler, Canada, June 2–5, ISBN 1-57735-199-1.

Ciravegna, F., Dingli, A., Petrelli, D. and Wilks, Y. (2002) 'User-system cooperation in document annotation based on information extraction', *EKAW '02: Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management*, Ontologies and the Semantic Web, Springer, Siguenza, Spain, pp.122–137.

Cohen, S., Mamou, J., Kanza, Y. and Sagiv, Y. (2003) 'Xsearch: a semantic search engine for xml', *VLDB 2003, Proceedings of 29th International Conference on Very Large Data Bases*, September 9–12, Berlin, Germany, pp.45–56.

Crestani, F. and Lee, P.L. (1999) 'WebSCSA: web search by constrained spreading activation', *Proceedings of IEEE ADL 99 – Advances in Digital Libraries Conference*, Baltimore, Maryland, USA, pp.163–170.

Davies, J. and Weeks, R. (2004) 'QuizRDF: search technology for the semantic web', *37th Hawaii International Conference on System Sciences (HICSS-37 2004), CD-ROM/Abstracts Proceedings*, 5–8 January, Big Island, HI, USA, IEEE Computer Society, ISBN 0-7695-2056-1.

Ding, H., Solvberg, I.T. and Lin, Y. (2004a) 'A vision on semantic retrieval in P2P network', *18th International Conference on Advanced Information Networking and Applications (AINA'04)*, Fukuoka, Kyushu, Japan, Vol. 1**,** pp.177–182.

Ding, L., Finin, T., Joshi, A., Peng, Y., Cost, R.S., Sachs, J., Pan, R., Reddivari, P., Doshi, V. and Reddivari, P. (2004b) 'Swoogle: a search and metadata engine for the semantic web', *Thirteenth ACM Conference on Information and Knowledge Management (CIKM'04)*, Washington DC, USA, pp.652–659.

Ehrig, M., Tempich, C., Broekstra, J., van Harmelen, F., Sabou, M., Siebes, R., Staab, S. and Stuckenschmidt, H. (2003) 'SWAP – ontology-based knowledge management with peer-to-peer technology', *Konferenz Professionelles Wissensmanagement*, Lucern, Luzern.

Fellbaum, C. (Ed.) (1998) *Wordnet: An Electronic Lexical Database*, MIT Press, ISBN-10:0-262-06197-XISBN-13: 978-0-262-06197-1.

Finin, T., Mayfield, J., Joshi, A., Cost, R.S. and Fink, C. (2005) 'Information retrieval and the semantic web', *System Sciences, HICSS'05. Proceedings of the 38th Annual Hawaii International Conference*, 03–06 January, pp.113a–113a, Digital Object Identifier 10.1109/HICSS.2005.319.

García, E. and Sicilia, M-Á. (2003) 'Designing ontology-based interactive information retrieval interfaces', *Workshop on Human Computer Interface for Semantic Web and Web Applications (HCI-SWWA)*, Lecture Notes in Computer Science 2889, Springer, New York, pp.152–165.

Glover, E.J., Lawrence, S., Gordon, M.D., Birmingham, W.P. and Giles, C.L. (2001) 'Web search – your way', *Communications of the ACM*, Vol. 44, No. 12, December, pp.97–102.

Gruber, T.R. (1993) 'A translation approach to portable ontology specifications', *Knowledge Acquisition*, Vol. 5, No. 2, pp.199–220.

Guarino, N., Masolo, C. and Vetere, G. (1999) 'Ontoseek: content-based access to the web', *Intelligent Systems and their Applications, IEEE*, Vol. 14, No. 3, May–June, pp.70–80.

Guha, R., McCool, R. and Miller, E. (2003) 'Semantic search', *WWW '03: Proceedings of the Twelfth International Conference on World Wide Web*, May, Budapest, Hungary.

Heflin, J. and Hendler, J. (2000) 'Searching the web with SHOE', *Artificial Intelligence for Web Search. Papers from the AAAI Workshop*, WS-00-01, AAAI Press, Menlo Park, CA, pp.35–40.

Horrocks, I. (2002) 'DAML+OIL: a reason-able web ontology language', *EDBT '02: Proceedings of the 8th International Conference on Extending Database Technology*, Springer, Prague, pp.2–13.

Hyvönen, E., Saarela, S. and Viljanen, K. (2003) 'Ontogator: combining view- and ontology-based search with semantic browsing', *Proceedings of XML Finland*, October 30–31, Kuopio, Finland, Paper presented at the *International SEPIA Conference*, Helsinki, September 18–20.

Jelmini, C. and Marchand-Maillet, S. (2004) 'Ontology reasoning for mulitmedia semantic retrieval', *Multimodal Interaction and Related Machine Learning Algorithms Workshop*, Martigny, Switzerland.

Karvounarakis, G., Alexaki, S., Christophides, V., Plexousakis, D. and Scholl, M. (2002) 'RQL: a declarative query language for RDF', *WWW'02: Proceedings of the Eleventh International Conference on World Wide Web*, Honolulu, Hawaii, USA.

Khan, L., McLeod, D. and Hovy, E.H. (2004) 'Retrieval effectiveness of an ontology-based model for information selection', *The VLDB Journal – The International Journal on Very Large Data Bases*, Vol. 13, No. 1, pp.71–85.

Kifer, M., Lausen, G. and Wu, J. (1995) 'Logical foundations of object-oriented and frame-based languages', *Journal of the ACM*, Vol. 42, No. 4, pp.741–843.

Labrou, Y. and Finin, T. (1999) 'Yahoo! as an ontology: using Yahoo! categories to describe documents', *CIKM '99: Proceedings of the Eighth International Conference on Information and Knowledge Management*, ACM Press, New York, NY, USA, pp.180–187.

Maedche, A., Motik, B., Stojanovic, L., Studer, R. and Volz, R. (2003) 'Ontologies for enterprise knowledge management', *IEEE Intelligent Systems*, Vol. 18, No. 2, pp.26–33.

Manola, F. and McBride, B. (2004) 'RDF primer, W3C recommendation 10 February 2004', *Technical Report*, W3C, http://www.w3.org/TR/rdf-primer/ (2004-07-15).

McGuiness, D.L. and van Harmelen, F. (2004) 'OWL web ontology language – overview – W3C recommendation 10 February 2004', *Technical Report*, W3C, http://www.w3.org/TR/owl-features/ (2004-7-15).

Mitra, M. and Chaudhuri, B.B. (2000) 'Information retrieval from documents: a survey', *Information Retrieval*, Vol. 2, Nos. 2–3, pp.141–163.

Mitra, M., Singhal, A. and Buckley, C. (1998) 'Improving automatic query expansion', *SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM Press, New York, NY, USA, pp.206–214.

Moench, E., Ullrich, M., Schnurr, H-P. and Angele, J. (2004) 'SemanticMiner – ontology-based knowledge retrieval', *Technical Report*, Ontoprise Gmbh, http://www.ontoprise.de/documents/SemanticMiner_Ontology-Based_Knowledge_Retrieval.pdf (2004-07-16).

Rocha, C., Schwabe, D. and de Aragao, M.P. (2004) 'A hybrid approach for searching in the semantic web', *WWW '04: Proceedings of the Thirteenth International Conference on World Wide Web*, New York, NY, USA, pp.374–383.

Seaborne, A. (2004) 'RDQL – a query language for RDF – W3C member submission 9 January 2004', *Technical Report*, W3C, http://www.w3.org/Submission/RDQL/ (2004-07-15).

Semagix (2005) *Freedom White Paper*, Semagix Ltd., http://wwwt.semagix.com/documents/SemagixFreedomWhitePaperUKV4_000.pdf, (2005-04-11).

Shah, U., Finin, T., Joshi, A., Cost, R.S. and Mayfield, J. (2002) 'Information retrieval on the semantic web', *CIKM '02: Proceedings of the Eleventh International Conference on Information and Knowledge Management*, ACM Press, McLean, Virginia, USA, pp.461–468.

Sheth, A., Bertram, C., Avant, D., Hammond, B., Kochut, K. and Warke, Y. (2002) 'Managing semantic content for the web', *IEEE Internet Computing*, Vol. 6, No. 4, pp.80–87.

Soo, W.V., Yang, Y.S., Chen, L.S. and Fu, T.Y. (2004) 'Ontology acquisition and semantic retrieval from semantic annotated Chinese poetry', *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries*, ACM Press, Tucson, Arizona, USA, pp.345–346.

Stojanovic, N. (2003) 'On analysing query ambiguity for query refinement: the librarian agent approach', *Conceptual Modeling – ER 2003, 22nd International Conference on Conceptual Modeling*, Chicago, IL, USA, October 13–16, *Proceedings*, pp.490–505.

Zhu, W-D., Dreves, G., Fichtinger, G., Bartlett, D., Myburgh, G. and Sreeraman, G. (2004) *Content Manager Implementation and Migration Cookbook*, IBM, http://www.redbooks.ibm.com/redbooks/pdfs/sg247051.pdf.

Note

[1]TREC – Text REtrieval Conference: http://trec.nist.gov/.