2010

An Ontology-Based Natural Language Service Discovery Engine – Design and Experimental Evaluation

Joerg Becker
ERCIS, joerg.becker@ercis.uni-muenster.de

Oliver Mueller
ERCIS, oliver.mueller@ercis.uni-muenster.de

Manuel Woditsch
ERCIS, manuel.woditsch@ercis.unimuenster.de

AN ONTOLOGY-BASED NATURAL LANGUAGE SERVICE DISCOVERY ENGINE – DESIGN AND EXPERIMENTAL EVALUATION

Journal:	*18th European Conference on Information Systems*
Manuscript ID:	ECIS2010-0305.R1
Submission Type:	Research-in-Progress Paper
Keyword:	Service oriented architecture (SOA), Semantic web, Service science, Information retrieval (IR)

AN ONTOLOGY-BASED NATURAL LANGUAGE SERVICE DISCOVERY ENGINE – DESIGN AND EXPERIMENTAL EVALUATION

Becker, Jörg, European Research Center for Information Systems, University of Münster, Leonardo-Campus 3, 48149 Münster, Germany, joerg.becker@ercis.uni-muenster.de

Müller, Oliver, European Research Center for Information Systems, University of Münster, Leonardo-Campus 3, 48149 Münster, Germany, oliver.mueller@ercis.uni-muenster.de

Woditsch, Manuel, European Research Center for Information Systems, University of Münster, Leonardo-Campus 3, 48149 Münster, Germany, manuel.woditsch@ercis.uni-muenster.de

Abstract

Around the globe, we are witnessing a transition from a primarily goods-based to a service-based economy. At the same time, advances in information technology provide significant opportunities for the electronic provisioning of services via the internet. As a result, more and more services are being traded online. Just as search tools are indispensable to discover information or goods on the web, they will become vital for discovering services on the web. In this paper we present Themis-S, a prototype of an ontology-based natural language service discovery engine. In a series of experiments, we evaluate the retrieval effectiveness of Themis-S in comparison to other state-of-the-art information retrieval models. The experiments indicate that Themis-S, incorporating WordNet as a general purpose ontology, can outperform systems applying syntactic information retrieval models such as the Vector Space Model (VSM) or the Probabilistic Relevance Model (PRM).

Keywords: service discovery, ontology, information retrieval, experiment

1 INTRODUCTION

Over the last decades, we have been witnessing a transition from a primarily goods-based to a more and more service-based economy in most developed countries. For example, services account for more than 80% of the current gross domestic product of the United States. At the same time, advances in information technology (e.g. broadband networks, rich user interfaces, electronic payment, or mobile devices) provide significant opportunities for the electronic provisioning of services via the internet (Rai & Sambamurthy 2006, Williams & Chatterjee & Rossi 2008). Some services, which allow for a fully digitized interaction between service provider and service consumer, can be completely delivered online, for example as web services, application programming interfaces, or rich internet applications. Other services, which require a physical interaction with the consumer, can still be discovered, requested, coordinated, or settled online.

Well-known online services from major players such as Google or Amazon are not isolated examples, but represent a major trend towards an ongoing digitization of services that “from every indication is likely to be more than a passing fad” (Williams et al. 2008). Just as search tools are indispensable to discover information or goods on the web, they will become vital for discovering services online. Already today, major online service marketplaces list tens of thousands service offerings and hence need sophisticated discovery mechanisms in order to be conveniently explorable by potential consumers.

Service discovery can be described as the task of matching the needs of a potential service consumer with the offerings of service providers. Existing service discovery approaches can be classified along several dimensions (see Garofalakis & Panagis & Sakkopoulos & Tsakalidis (2006) for a detailed discussion). Approaches stemming from the semantic web community aim at enabling software systems to “understand” the capabilities and conditions of a service by using formal languages to enrich service descriptions and corresponding queries through semantic annotations. These approaches allow for a precise, unambiguous, and machine-processable description of services and service requests. However, this comes at the expense of using artificial languages and formal ontologies to communicate and reason about services. Prominent examples of such languages and ontologies include OWL-S, WSMO, and SAWSDL. However, so far the diffusion of semantic web services is modest, primarily because the required efforts for building appropriate formalizations are enormous (Haniewicz & Kaczmarek & Zyskowski 2008). Approaches originating from the information retrieval community deliberately abstain from such formalizations and rely on natural language descriptions of services. The standard Boolean model for matching queries in the form of keywords with natural language descriptions of services is the most common method. Prominent examples include keyword-style search functions at service marketplaces like programmableweb.com or seekda.com. However, most of these approaches are solely based on syntax matching and thus do not take into account the polysemy of natural language.

Against this background, the aim of our research is to develop an easy-to-use approach to service discovery, which can be regarded as a meet-in-the-middle between the application of heavyweight semantic web technologies and syntactic information retrieval models. We chose to build our discovery engine upon the enhanced Topic-based Vector Space Model (eTVSM), an information retrieval model which is able to consider semantic relations such as synonymy, homonymy, and hyponymy/hypernymy in natural language text by exploiting the lexical semantics of an underlying ontology.

In this paper we will explain how eTVSM can be applied for service discovery and compare its effectiveness, i.e. the ability to retrieve relevant and only relevant results matching a specific information need, to two syntax-based state-of-the-art information retrieval models, namely the Vector Space Model (VSM) and the Probabilistic Relevance Model (PRM).

Hence, our research is driven by the following research question:

- In terms of retrieval effectiveness, can an eTVSM-based service discovery engine outperform discovery engines using syntax-based state-of-the-art information retrieval models (i.e. VSM and PRM)?

The remainder of this paper is structured as follows: In Section 2 we present the idea behind using eTVSM for service discovery, explain the model in detail, and illustrate how WordNet can be used as an underlying ontology. In Section 3 we summarize the setup of our evaluation experiments using a test collection of geo web services and present the quantitative results of the experiments. In Section 4 we take a look at related work that is conceptually similar to our approach. Finally, in Section 5 we sum up the conclusions of our work, expose its limitations, and give an outlook for further research.

2 THEMIS-S – AN ONTOLOGY-BASED NATURAL LANGUAGE SERVICE DISCOVERY ENGINE

Our attempts to apply eTVSM to service discovery are motivated by the observation that service matchmaking is hampered by what has been called the vocabulary problem (Furnas & Landauer & Gomez & Dumais 1987). The user is looking for a service but is unsure what keywords to use to describe for the desired service or, respectively, its operations, inputs, and outputs. In their seminal work Furnas et al. (1987) have shown that using a single access term for an operation or data object results in very poor hit rates of 10-20%. “There is no one good access term for most objects. The idea of an ‘obvious’, ‘self-evident’, or ‘natural’ term is a myth” (Furnas et al. 1987). Providing aliases such as synonyms, hyponyms, or hypernyms can significantly improve the performance to hit rates above 50%. The vocabulary problem is common to all retrieval systems that are based on natural language. However, we argue that it is especially severe in service discovery, as service descriptions mainly comprise of technical terms chosen by a developer and contain only very sparse meta-data or documentation. Hence, the probability that a query term of a requestor exactly matches the name of a service, operation, input, or output is low.

2.1 The Enhanced Topic-Based Vector Space Model (eTVSM)

eTVSM can consider lexical semantics when processing documents and queries. The rationale behind using eTVSM for service discovery is straightforward. Both, service descriptions of providers as well as service requests of potential consumers are formulated as free text natural language documents. When the discovery engine parses these documents it tries to identify terms and extracts related meanings along with synonyms, hyponyms, and hypernyms that are inherent in an underlying ontology. From the set of extracted concepts a document model is constructed which acts as a representation of the original full text document for the subsequent steps of calculating similarities and ranking results.

The basic idea of eTVSM is rooted in the classic Vector Space Model (Salton & Wong & Yang 1975). eTVSM represents documents as vectors in a vector space. The similarity between documents is expressed by the angle between their document vectors. However – compared to VSM – eTVSM provides some significant modifications (Polyvyanyy & Kuropka 2007). The main advancement is, that eTVSM does not suffer from the false assumption that two different terms are independent (orthogonal) of each other. eTVSM operates on concepts (word meanings) rather than terms (words). Semantic relations (i.e. synonymy, homonymy, and hyponymy/hypernymy) between these concepts can be modelled in a domain ontology. When constructing document vectors and calculating their similarity, eTVSM exploits these semantic relations. The basic assumption is, that the shorter the distance (path) between two concepts (nodes) in the ontology (graph) is, the higher is their semantic similarity. Measuring and combining the semantic similarity between all concepts inherent in a query and a document leads to an overall similarity score. First evaluations with standard test collections have shown that this approach can dramatically improve retrieval effectiveness (Knackstedt & Kuropka & Müller & Polyvyanyy 2008, Polyvyanyy et al. 2007).

To build an appropriate domain ontology, eTVSM offers three concepts: topics, interpretations, and terms (see entity-relationship diagram in Fig. 1). These concepts are organized in a hierarchical, non-cyclic directed graph structure. Edges of the graph aim to specify semantic relations between concepts. Topics are the most general semantic entities of an eTVSM ontology. All concepts connected to a topic are considered on-topic, i.e. they are in scope of the current topic. Topic relations are expressed in a topic map. A topic map is a directed graph with topics as nodes. Graph edges assign super-sub-topic relations which are meant to represent hyponymy/hypernymy relations in lexical semantics. A graph consisting of jointly connected topics represents a distinct subset of the domain of discourse. Within this graph all topics gain some level of similarity based on the amount of intermediate topics in the topic map structure. Interpretations are links between topics and terms. Conceptually, interpretations represent the meaning of terms. By introducing this intermediate concept the modeller of an ontology receives more freedom and opportunities to express semantic phenomena. For example, mapping two terms to the same interpretation expresses total synonymy. Mapping one term to two or more interpretations expresses homonymy. Every interpretation is linked to exactly one topic. Links between interpretations are not allowed. Terms are treated as the smallest unit of information that represents one or several (in case of homonyms) meanings. To express this multiplicity in meaning terms might be linked to an arbitrary number of interpretations. Such a link should be further enriched by so-called support terms. Support terms are terms that frequently co-occur with a specific term. They are intended to be used for disambiguation of term-interpretation links, i.e. to come to a proper decision of what is the “right” interpretation of a term (in case it might have several ones).

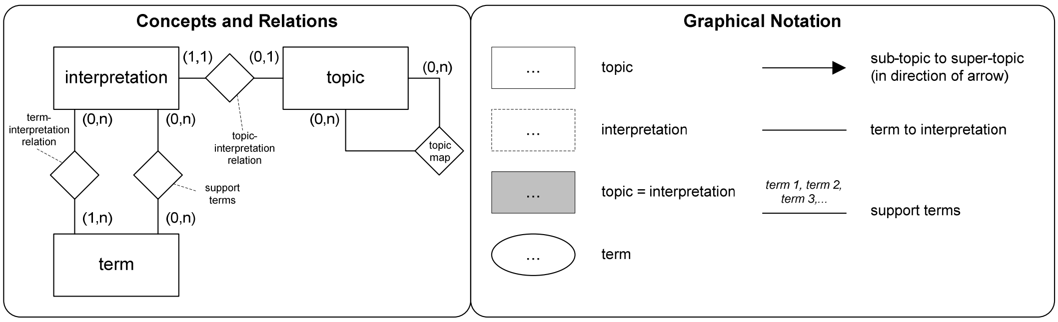


Fig. 1. Concepts, relations and graphical notation of an eTVSM ontology

In the following, we will introduce the formalisms required to represent the structure of an eTVSM ontology (equations (1) – (6)), to construct topic and interpretation vectors (equations (7) – (11)), and to build document vectors and calculate similarities between documents (equations (12) – (14)).

A set of all topics is defined by:

$$\theta = \{\tau_1, \tau_2, \ldots, \tau_{\#\theta}\} \tag{1}$$

In order to represent the structure of a topic map, we define a super-topic relation $S(\tau_i)$ which provides a set of all direct parent topics of a topic τ_i:

$$S(\tau_i) \subseteq (\theta \setminus \tau_i) \tag{2}$$

From the super-topic relation $S(\tau_i)$ we can derive a *p*-level super-topic relation $S^p(\tau_i)$. This transitive relation provides super-topics that are *p* levels above the target topic:

$$\begin{aligned} S^{p}(\tau_i) &= S(\tau_i) && for \quad p = 1 \\ S^{p}(\tau_i) &= \bigcup_{\tau_k \in S^{p-1}(\tau_i)} S(\tau_k) && for \quad p > 1 \end{aligned} \tag{3}$$

To obtain all super-topics of a target topic we can use an unbound transitive super-topic relation $S^*(\tau_i)$:

$$S^*(\tau_i) = S^1(\tau_i) \cup S^2(\tau_i) \cup S^3(\tau_i) \cup \ldots \tag{4}$$

A set of leaf topics θ_L contains all topics that are not included in any super-topic relation of any topic from a topic map:

$$\theta_L = \{\tau_i \in \theta : \neg \exists \tau_k \in \theta \;\; with \;\; \tau_i \in S(\tau_k)\} \tag{5}$$

Complementary to θ_L is a set of internal topic nodes θ_N which comprises topics that have at least one sub-topic:

$$\theta_N = \theta \setminus \theta_L \tag{6}$$

Each topic τ_i posses a topic vector $\vec{\tau}_i$ with t dimensions. This vector defines the pairwise similarities between the topic τ_i and all other topics of the topic map. The approach for gaining the value of a topic vector is twofold:

In case of leaf topics, the corresponding dimension entry of a topic vector is set to 0 if two topics are un-connected and to 1 if a topic belongs to the set of super-topics of the target topic or is identical to the topic itself. Afterwards, the topic vector is normalized to the length of 1:

$$\forall \tau_i \in \theta_L : \vec{\tau}_i = \left| \left(\tau^*_{i,1}, \tau^*_{i,2}, \ldots, \tau^*_{i,\#\theta} \right) \right| \; with \; \tau^*_{i,k} = \begin{cases} 1 \; if \; \tau_k \in S^*(\tau_i) \vee i = k \\ 0 \, else \end{cases} \tag{7}$$

In case of internal topics, topic vectors are obtained as the sum of the topic vectors of all direct sub-topics that is normalized to the length of 1:

$$\forall \tau_i \in \theta_N : \vec{\tau}_i = \left| \sum_{\tau_S \in \theta :\, \tau_i \in S(\tau_S)} \vec{\tau}_S \right| \tag{8}$$

Once we have obtained all topic vectors, we can calculate the similarity of two topics as the scalar product of their topic vectors. Because all topic vectors are normalized, the scalar product is equal to the cosine of the angle between the two topic vectors:

$$sim(\tau_i, \tau_j) = \vec{\tau}_i \vec{\tau}_j = \sum_{k=1}^{\#\theta} \tau_{i,k} \tau_{j,k} \tag{9}$$

A set of all interpretations is given by:

$$\Phi = \{\phi_1, \phi_2, \ldots, \phi_{\#\Phi}\} \tag{10}$$

As there exists a one-to-one relationship between interpretations and topics, the similarity between two interpretations is equal to the similarity of their topics:[1]

$$sim(\phi_i, \phi_j) = sim(\tau_i, \tau_j) \tag{11}$$

To calculate similarities between documents a document vector $\vec{d_j}$ for each document $d_j \in D$ is constructed. This document vector represents a document as a bag of interpretations which can be found in the document (via the terms related to those interpretations). It also comprises the weight ω_{d_j,ϕ_i} of an interpretation ϕ_i in a document d_j, which might be a simple occurrence count or a more elaborate weighting scheme such as a combination of term frequency and inverse document frequency (tf-idf weighting). Hence, the document vector is the normalized weighted sum of the interpretation vectors of the interpretations present in the document and is calculated as follows:

$$\forall d_j \in D : \vec{d_j} = \frac{1}{\left|\vec{\delta_j}\right|}\vec{\delta_j} \quad with \quad \vec{\delta_j} = \sum_{\phi_i \in \Phi} \omega_{\phi_i, d_j} \vec{\phi_i} \tag{12}$$

The document vector length is calculated as the scalar product of the interpretation vectors of all interpretations present in the document and is obtained as:

$$\left|\vec{\delta_j}\right| = \left|\sum_{\phi_k \in \Phi} \omega_{d_j,\phi_k} \vec{\phi_k}\right| = \sqrt{\left|\sum_{\phi_k \in \Phi} \omega_{d_j,\phi_k} \vec{\phi_k}\right|^2} = \sqrt{\sum_{\phi_k \in \Phi} \sum_{\phi_l \in \Phi} \omega_{d_j,\phi_k} \omega_{d_j,\phi_l} \vec{\phi_k}\vec{\phi_l}} \tag{13}$$

Finally, the similarity between two documents d_i and d_j is obtained as the scalar product of their document vectors. Considering the normalization, the similarity value is equal to the cosine of the angle between the two document vectors:

$$sim(d_i, d_j) = \vec{d_i}\vec{d_j} = \frac{1}{\left|\vec{\delta_i}\right|}\vec{\delta_i} \frac{1}{\left|\vec{\delta_j}\right|}\vec{\delta_j} \tag{14}$$

2.2 Using WordNet as an underlying Ontology

As it becomes clear from the above explanations, the application of eTVSM requires an ontology, which specifies the lexical semantics of the domain of discourse. In contrast to the logic-based formal ontologies used for semantic web service approaches like OWL-S or WSMO, our approach merely needs a semi-formal, thesaurus-like ontology that comprises the natural language terms of the domain of discourse as well as the concepts they represent and the linguistic relations between those concepts. WordNet is a possible candidate for such an ontology.

WordNet is a lexical database of the English language developed by a team of psychologists and linguists at Princeton University (Miller 1995). WordNet 3.0 currently comprises about 155.280 different words (nouns, verbs, adjectives, and adverbs) which are grouped into some 117.650 sets of cognitive synonyms, each expressing a distinct concept. These sets of synonyms are called synsets, each having a natural language definition called gloss. Synonymous words are mapped to the same synset; homonymous words are mapped to more than one synset. Synsets are linked by different

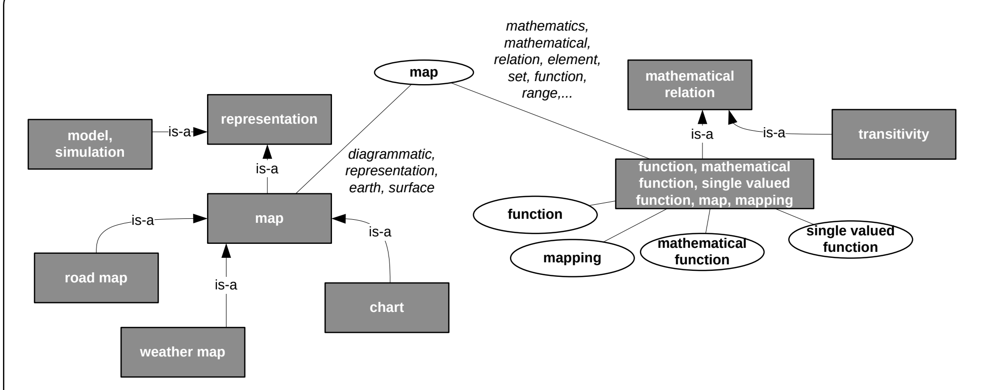
[1] Variants of eTVSM allow many-to-many relationships between interpretations and topics. In this case, the similarity of two interpretations is equal to the normalized and weighted sum of the topic vectors of all topics related to an interpretation.

semantic relations, e.g. hyponymy/hypernymy (sometimes called generalization/specialization, subset/superset or is-a relation).

WordNet

- S: (n) **map** (a diagrammatic representation of the earth's surface (or part of it))
  - *direct hyponym* / *full hyponym*
    - S: (n) chart (a map designed to assist navigation by air or sea)
    - S: (n) choropleth map (a map that uses graded differences in shading or color or the placing of symbols inside defined areas on the ma; property or quantity in those areas)
    - S: (n) contour map, relief map (a map having contour lines through points of equal elevation)
    - S: (n) plat (a map showing planned or actual features of an area (streets and building lots etc.))
    - S: (n) road map (a map showing roads (for automobile travel))
    - S: (n) sketch map (a map drawn from observation (rather than from exact measurements) and representing the main features of an area
    - S: (n) weather map, weather chart ((meteorology) a map showing the principal meteorological elements at a given time and over an ext
  - *direct hypernym* / *inherited hypernym* / *sister term*
    - S: (n) representation (a creation that is a visual or tangible rendering of someone or something)
  - *derivationally related form*
- S: (n) function, mathematical function, single-valued function, **map**, mapping ((mathematics) a mathematical relation such that each element of a giver element of another set (the range of the function))
  - *direct hyponym* / *full hyponym*
  - *domain category*
  - *direct hypernym* / *inherited hypernym* / *sister term*
    - S: (n) mathematical relation (a relation between mathematical expressions (such as equality or inequality))

eTVSM ontology



eTVSM similarity matrix

	representation	model, simulation	map	road map	weather map	chart	mathematical relation	function, ...	transitivity
representation	1,000	0,855	0,855	0,754	0,754	0,754	0,000	0,000	0,000
model, simulation	0,855	1,000	0,463	0,408	0,408	0,408	0,000	0,000	0,000
map	0,855	0,463	1,000	0,882	0,882	0,882	0,000	0,000	0,000
road map	0,754	0,408	0,882	1,000	0,667	0,667	0,000	0,000	0,000
weather map	0,754	0,408	0,882	0,667	1,000	0,667	0,000	0,000	0,000
chart	0,754	0,408	0,882	0,667	0,667	1,000	0,000	0,000	0,000
mathematical relation	0,000	0,000	0,000	0,000	0,000	0,000	1,000	0,866	0,866
function, ...	0,000	0,000	0,000	0,000	0,000	0,000	0,866	1,000	0,866
transitivity	0,000	0,000	0,000	0,000	0,000	0,000	0,866	0,866	1,000

Fig. 2. Transformation of WordNet into an eTVSM ontology and similarity matrix (exemplary excerpt)

Although WordNet was not designed to be an ontology in the computer science sense and a number of problems may arise when using it as a formal ontology (e.g. confusion between concepts and individuals, heterogeneous levels of abstraction (Gangemi & Guarino & Oltramari 2001)), numerous projects successfully applied WordNet as a semi-formal ontology to support information retrieval tasks.

Using the notation introduced in Figure 1, Figure 2 illustrates how the word ‘map’, including the different synsets to which ‘map’ is linked, as well as the related synsets (hyponyms/hypernyms) and terms are transformed from WordNet. The matrix at the bottom of Figure 2 shows the corresponding similarity values calculated according to equations (7) – (11).

3 EXPERIMENTAL EVALUATION OF THEMIS-S

To evaluate the retrieval effectiveness of Themis-S, we follow the traditional way of evaluating information retrieval systems as established during the TREC ad hoc retrieval experiments (Voorhees & Harman 2005). In the following sections we will explain the setup of our experiments and report on the results.

3.1 Proposition and Variables

The aim of our experiments is to test whether eTVSM can be used as an information retrieval model for service discovery. Hence, the experiments are based on the following research proposition:

- Due to the consideration of lexical semantics, an eTVSM-based service discovery engine can outperform discovery engines using syntax-based state-of-the-art information retrieval models (i.e. VSM and PRM).

The independent variable of our experiments is the information retrieval model. We test three different models:

- eTVSM tf-idf: The enhanced Topic-based Vector Space Model with a standard tf-idf weighting function, as it is described in section 2 of this paper. Due to the use of an ontology eTVSM can consider lexical semantics.
- VSM tf-idf: The classic Vector Space Model with a standard tf-idf weighting function. A prominent implementation of VSM is the SMART system, which is consistently among the best-performing systems of the TREC ad hoc retrieval experiments (Voorhees et al. 2005). VSM is syntax-based and does not consider semantics.
- OKAPI BM25: OKAPI is a retrieval system based on a Probabilistic Relevance Model (PRM) with the inherent BM25 weighting function (Sparck Jones & Walker & Robertson 2000). OKAPI is consistently among the best-performing systems of the TREC ad hoc retrieval experiments (Voorhees et al. 2005). OKAPI is syntax-based and does not consider semantics.

To build actual retrieval systems embodying the above models we used two open source projects: The Themis Framework (http://code.google.com/p/ir-themis/), which implements eTVSM tf-idf and VSM tf-idf, and the Lemur Toolkit (http://www.lemurproject.org/), which implements (besides others) the OKAPI BM25 model. All systems use the Porter algorithm stemmer for stemming word forms (Porter 1980).

The dependent variables of our experiments are several standard retrieval effectiveness evaluation measures. Retrieval effectiveness describes the capability of a system to retrieve those documents that are actually relevant to a certain information need. The basic components of retrieval effectiveness are recall, i.e. the ability to retrieve *all* relevant documents, and precision, i.e. the ability to retrieve *only* relevant documents. Based on recall and precision a number of composite measures can be calculated. In the long tradition of experimentation in the information retrieval discipline the following measures evolved as standard evaluation measures (Buckley & Voorhees 2005):

- Eleven-Point Average Precision (EPAP): The average of precision at eleven standard recall points (recall levels 0.0 to 1.0). These values can be plotted to give a precision-recall curve.

- R-Precision: Precision after *R* documents have been retrieved, where *R* is the number of relevant documents for an information need. R-Precision has proven to be a good measure of overall system effectiveness.
- Mean Average Precision (MAP): The precision at each relevant document, averaged over all relevant documents for an information need. This is the measure most often used to characterize the overall effectiveness of a retrieval system.

3.2 Test Collection

In order to evaluate the effectiveness of the three retrieval models a test collection is needed. A test collection comprises three components: a document collection, information needs including queries, and relevance judgments (Buckley et al. 2005). Normally, information retrieval experiments are conducted with standard test collection such as the TREC or REUTERS collections. However, as we are not interested in general purpose information retrieval but service discovery, we need a test collection consisting of service descriptions (document collection), service requests (information needs including queries), and corresponding relevance judgments. Surprisingly, despite the huge efforts put into the development of service discovery approaches in the semantic web service community, only few and small test collections for the evaluation of those approaches exist. Additionally, the available test collections (see Küster & König-Ries (2008) for an overview) focus on formal descriptions of services and queries and only contain little natural language text. Hence, we were forced to assemble our own test collection.

Our document collection contains 100 descriptions of geo web services in English natural language text, ranging from 74 to 1,271 words in length. In a first step, service descriptions were gathered by consulting two well known service repositories, www.programmableweb.com and www.seekda.com. In a second step, the services were categorized according to their capabilities to later derive suitable information needs and queries.

Based on the results of the categorization we identified two suitable information needs, which we distributed to a group of information systems students at a large German university:

- Information need #1: Find a service that returns the geographic location of an IP address.
- Information need #2: Find a service that provides a dynamic map.

The actual descriptions of the information needs were much longer than presented here and formulated in German language to avoid any kind of bias caused by our wording. The students were asked to formulate an English query in keyword style which they would use to search for appropriate services using a special search engine or service repository. For each information need we randomly selected the queries of 30 individuals. The resulting 60 queries ranged in length from 2 to 10 words. Because we derived the information needs from the categorization of our document collection, we were able to create binary relevance judgments for all information need-document pairs. For information need #1 there were 8 relevant service descriptions in the document collection and for information need #2 there were 14 relevant service descriptions in the collection.

3.3 Measurements and Evaluation

Figure 3 presents the interpolated precision-recall plots for the three tested information retrieval models. The graphs have been produced by calculating, for each information retrieval model, the arithmetic mean of the average of precision at the eleven standard recall points (0.0 to 1.0) (EPAP) of information need #1 and #2. At all eleven standard recall points Themis-S, i.e. eTVSM tf-idf, dominates the other two systems.

As EPAP is known to be an unreliable overall system effectiveness measure, we also calculated the R-Precision and MAP values of the three retrieval models for both information needs. Table 1 gives an overview of these values and also shows the arithmetic mean of the evaluation measures over the two information needs. For information need #1 OKAPI BM25 performs better than the other two systems; for information need #2 Themis-S is at the top. Across both information needs Themis-S turns out to be the best-performing system.

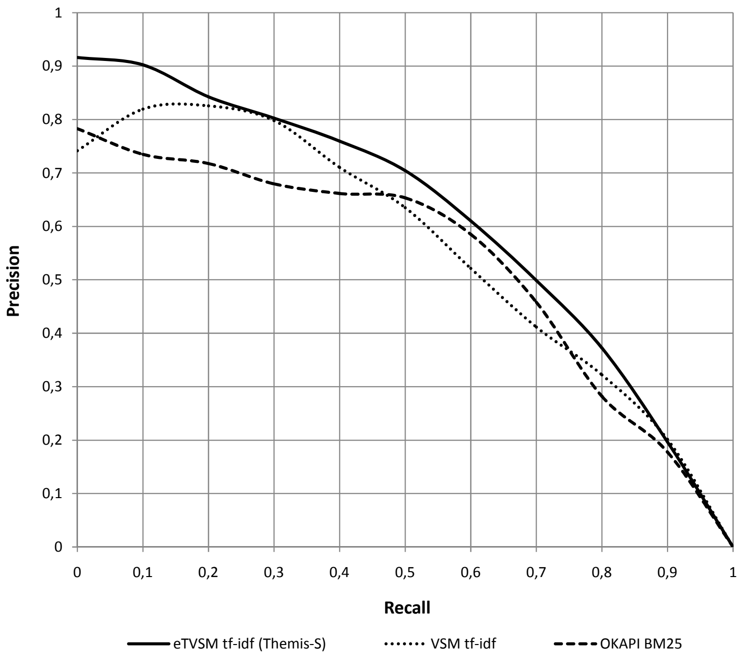


Fig. 3. Interpolated precision-recall plots

The data supports our research proposition. Both, the precision-recall plots as well as the R-Precision and MAP values indicate, that Themis-S can outperform the systems based on VSM tf-idf and OKAPI BM25 in terms of retrieval effectiveness. In comparison to VSM tf-idf an improvement of 6.39% for R-Precision and 9.12% for MAP could be realized. In comparison to OKAPI BM25 an improvement of 14.14% for R-Precision and 11.47% for MAP has been reached. As a way of gauging the importance of these experimental results, we refer to Sparck Jones (1974), who states that improvements of 5% are noticeable and improvements of 10% are material.

Table 1. Evaluation measurements

	Measure	eTVSM tf-idf (Themis-S)	VSM tf-idf	OKAPI BM25
Information need #1	R-Precision	0.671	0.613	0.688
	MAP	0.759	0.700	0.763
Information need #2	R-Precision	0.579	0.562	0.407
	MAP	0.638	0.580	0.490
Mean	R-Precision	0.625	0.587	0.547
	MAP	0.699	0.640	0.627

4 RELATED WORK

In the last years, a shift from service matchmaking approaches applying formal logical inference techniques towards more similarity-based algorithms has been observed (Fernandez & Hayes & Loutas & Peristeras 2008). Consequently, several approaches comparable to the one described in this study have been published. Platzer & Dustdar (2005) have build a search engine for web services based on the standard Vector Space Model (VSM). Unfortunately, they do not provide any details on the effectiveness of their system. Some matchmaking engines like the systems of Dong & Halevy & Madhavan & Nemes & Zhang (2004) and Hao & Cao & Zhang (2009) apply clustering and schema matching algorithms on WSDL service descriptions. The former system, called Woogle, has been evaluated in terms of retrieval effectiveness and outperformed various naïve methods. Numerous web search engine prototypes apply latent semantic indexing (LSI) techniques to identify relationships and meanings of terms in natural language service descriptions without the need for an underlying ontology (e.g. Ismaili & Zenuni & Raufi 2009, Ma & Zhang & He 2008, Sajjanhar & Hou & Zhang 2004, Vouros & Dimitrokallis & Kotis 2008). The respective authors report on significant improvements in comparison to simple Boolean search, but have not yet compared their models to more sophisticated information retrieval or semantic web approaches. A relatively new class of systems uses collaborative tagging, as known from popular Web 2.0 applications, for service matchmaking (Fernandez et al. 2008, Gawinecki & Cabri & Paprzycki & Ganzha 2010). These systems produced promising results in first experimental evaluations.

5 CONCLUSION, LIMITATIONS AND OUTLOOK

This paper introduced a novel approach to service discovery based upon eTVSM, which can be regarded as a meet-in-the-middle between the application of heavyweight semantic web technologies and syntactic information retrieval models. We presented Themis-S, a prototype of a service discovery engine implementing eTVSM in combination with WordNet as a possible ontology. In a series of experiments we evaluated the retrieval effectiveness of Themis-S in comparison to two systems applying syntax-based state-of-the-art information retrieval models, namely VSM and OKAPI. The domain chosen for the experiments was the area of geo web services. Due to a lack of appropriate standard test collections, the experiments were preformed on a newly assembled test collection, comprising 100 natural language service descriptions, two information needs with an overall of 60 queries, and two sets of relevance judgments.

The results indicate that Themis-S can outperform systems based on VSM and OKAPI. Improvements in retrieval effectiveness between 6% and 14% have been realized. For practice, the results can inform the developers of specialized search engines and service repositories. However, the relative small size of our test collection constitutes a potential threat to the external validity of our experiments. While more than 50 queries and improvements of more than 5% are said to be reliable dimensions, our document collection might be too small to draw valid generalizations from our results. This represents a general problem in the area of service discovery as bigger appropriate test collections do not exist. Hence, future research should focus on creating reasonable sized standards test collections for the evaluation of service discovery methods. Besides building a bigger test collection, our future research will focus on training Themis-S on the vocabulary of the services domain by customizing and expanding the underlying WordNet-based ontology. In addition, our interest is on comparing the retrieval effectiveness of our system to discovery engines applying semantic web technologies. First such comparisons will be carried out at the upcoming Semantic Service Selection (S3) contest.

References

Buckley, C. and Voorhees, E. M. (2005). Retrieval system evaluation. In TREC: Experiment and Evaluation in Information Retrieval (Voorhees, E. M. and Harman, D. K. Eds.), pp. 53–75, MIT Press, Cambridge.

Dong, X., Halevy, A., Madhavan, J., Nemes, E. and Zhang, J. (2004). Similarity Search for Web Services. In Proceedings of the 30th VLDB Conference, Toronto.

Fernandez, A., Hayes, C., Loutas, N. and Peristeras, V. (2008). Closing the Service Discovery Gap by Collaborative Tagging and Clustering Techniques. In Proceedings of the 2nd International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web (SMR2), Karlsruhe.

Furnas, G. W., Landauer, T. K., Gomez, L. M. and Dumais, S. T. (1987). The Vocabulary Problem in Human-System Communication. Communications of the ACM, 30 (11), pp. 964-971.

Gangemi, A., Guarino, N. and Oltramari, A. (2001). Conceptual analysis of lexical taxonomies: The case of wordnet top-level. In Proceedings of the Proceedings of the International Conference on Formal Ontology in Information Systems, pp. 285-296, Ogunquit.

Garofalakis, J., Panagis, Y., Sakkopoulos, E. and Tsakalidis, A. (2006). Contemporary Web Service Discovery Mechanisms. Journal of Web Engineering, 5 (3), pp. 265-265.

Gawinecki, M., Cabri, G., Paprzycki, M. and Ganzha, M. (2010). WSCOLAB: Structured Collaborative Tagging for Web Service Matchmaking. In Proceedings of the 6th International Conference on Web Information Systems and Technolgies (WEBIST), Valencia.

Haniewicz, K., Kaczmarek, M. and Zyskowski, D. (2008). Semantic Web Services Applications - a Reality Check. Wirtschaftsinformatik, 50 (1), pp. 39-39.

Hao, Y., Cao, J. and Zhang, Y. (2009). Efficient IR-Style Search over Web Services. In Proceedings of the 21st International Conference on Advanced Information Systems (CAiSE), Amsterdam.

Ismaili, F., Zenuni, X. and Raufi, B. (2009). A novel approach for efficiently finding Web Services on the Web. In Proceedings of the 31st International Conference on Information Technology Interfaces (ITI), Cavtat.

Knackstedt, R., Kuropka, D., Müller, O. and Polyvyanyy, A. (2008). An Ontology-based Service Discovery Approach for the Provisioning of Product-Service Bundles. In Proceedings of the 16th European Conference on Information Systems (ECIS), Galway.

Küster, U. and König-Ries, B. (2008). Towards Standard Test Collections for the Empirical Evaluation of Semantic Web Service Approaches. International Journal of Semantic Computing, 2 (57), pp. 1-22.

Ma, J., Zhang, Y. and He, J. (2008). Web Services Discovery Based on Latent Semantic Approach. In Proceedings of the IEEE International Conference on Web Services (ICWS), Bejing.

Miller, G. A. (1995). WordNet: a lexical database for English. Communications of the ACM, 38 (11), pp. 39-41.

Platzer, C. and Dustdar, S. (2005). A Vector Space Search Engine for Web Services. In Proceedings of the 3rd IEEE European Conference on Web Services, Växjö.

Polyvyanyy, A. and Kuropka, D. (2007). A Quantitative Evaluation of the Enhanced Topic Based Vector Space Model. Technical Report of the HPI No. 19.

Porter, M. F. (1980). An algorithm for suffix stripping. Program: electronic library and information systems, 14 (3), pp. 130-137.

Rai, A. and Sambamurthy, V. (2006). Editorial Notes - The Growth of Interest in Services Management: Opportunities for Information Systems Scholars. Information Systems Research, 17 (4), pp. 327-331.

Sajjanhar, A., Hou, J. and Zhang, Y. (2004). Algorithm for Web Services Matching. LNCS, 3007, pp. 665-670.

Salton, G., Wong, A. and Yang, C. S. (1975). A Vector Space Model for Automatic Indexing. Communications of the ACM, 18 (11), pp. 613-620.

Sparck Jones, K. (1974). Automatic indexing. Journal of Documentation, 30 (4), pp. 393-432.

Sparck Jones, K., Walker, S. and Robertson, S. E. (2000). A Probabilistic Model of Information Retrieval: Development and Comparative Experiments, Parts 1 and 2. Information Processing and Management, 36 (6), pp. 779-840.

Voorhees, E. M. and Harman, D. K. (2005). TREC: Experiment and evaluation in information retrieval. MIT Press, Cambridge.

Vouros, G. A., Dimitrokallis, F. and Kotis, K. (2008). Look Ma, No Hands: Supporting the Semantic Discovery of Services without Ontologies. In Proceedings of the 2nd International Workshop on Service Matchmaking and Resource Retrieval in the Semantic Web (SMR2), Karlsruhe.

Williams, K., Chatterjee, S. and Rossi, M. (2008). Design of Emerging Digital Services: a Taxonomy. European Journal of Information Systems, 17 (5), pp. 505-517.