The VLDB Journal (2003) / Digital Object Identifier (DOI) 10.1007/s00778-003-0100-6

THESUS: Organizing Web document collections based on link semantics

Maria Halkidi[1], **Benjamin Nguyen**[2], **Iraklis Varlamis**[1], **Michalis Vazirgiannis**[1]

[1] Athens University of Economics and Business, 76 Patision Street, Athens, Greece
e-mail: {mhalk, varlamis, mvazirg}@aueb.gr
[2] INRIA, Domaine de Voluceau, 78153 Le Chesnay, France
e-mail: Benjamin.Nguyen@inria.fr

Edited by ♣. Received: ♣/ Accepted: ♣
Published online: ♣♣ 2003 –

Abstract. The requirements for effective search and management of the WWW are stronger than ever. Currently Web documents are classified based on their content not taking into account the fact that these documents are connected to each other by links. We claim that a page's classification is enriched by the detection of its incoming links' semantics. This would enable effective browsing and enhance the validity of search results in the WWW context. Another aspect that is underaddressed and strictly related to the tasks of browsing and searching is the similarity of documents at the semantic level. The above observations lead us to the adoption of a hierarchy of concepts (ontology) and a thesaurus to exploit links and provide a better characterization of Web documents. The enhancement of document characterization makes operations such as clustering and labeling very interesting. To this end, we devised a system called THESUS. The system deals with an initial sets of Web documents, extracts keywords from all pages' incoming links, and converts them to semantics by mapping them to a domain's ontology. Then a clustering algorithm is applied to discover groups of Web documents. The effectiveness of the clustering process is based on the use of a novel similarity measure between documents characterized by sets of terms. Web documents are organized into thematic subsets based on their semantics. The subsets are then labeled, thereby enabling easier management (browsing, searching, querying) of the Web. In this article, we detail the process of this system and give an experimental analysis of its results.

Keywords: World Wide Web – Link analysis – Similarity measure – Document clustering – Link management – Semantics

1 Introduction

Various fields of science, from sociology to physics, share the common paradigm that an entity is usually *more than the sum of its parts*. However, when applied to searches on the Web, the traditional process is to consider that a page is defined exclusively by its content. On the one hand, World Wide Web surfers traditionally submit queries to search engines by using one or more keywords to define their interests. On the other hand, results are equally hard to tally since search engines usually return huge lists of URLs, most of which can be judged almost irrelevant to the query.

In such search engines (Google, Yahoo, Altavista), a Web page's semantics are derived from the keywords that are assigned to the page, usually because they appear in the body of the page. Yet the World Wide Web is a graph. More precisely still, it is a *directed labeled graph*, where the pages represent the nodes and links in the pages represent the edges of the graph. In this article, we will show how to add semantic labels to the edges of the graph of the Web and query the results. We believe that we will be able to extract crucial semantic material about a Web page from the pages that *point to* that page. This is what distinguishes the World Wide Web from a simple collection of documents. Links are often cornerstones in Web design and as such deserve to be taken into account when answering queries. We want to group Web documents into thematic subsets called THESUs (Thematic Subsets of the WWW). The semantic proximity of the pages in a THESU is not only derived by the pages' content but by the semantics of the links pointing to this page's (**link semantics**). This aspect is generally ignored in popular search engines.

Note: THESUS is the name of the system. A THESU is a thematic subset, and the plural is THESUs. We refer indifferently to a document, a page, a Web resource, or a node when defining the basic entity of our system.

THESUS is NOT a search engine but rather a system that enhances the semantic organization of a set of pages and guides the user within this set. It is envisioned as a personal service that will assist the user in the creation and querying of rather compact high-quality thematic collections of pages. Of course, the concepts developed here are expandable to the full WWW; we bore scalability in mind during the creation and the implementation of THESUS.

2 Motivation

Link use. As already mentioned, the World Wide Web is a directed labeled graph whose edges carry information from the source node to the target node. Though link analysis and link structures are exploited in order to rank the importance of pages [23], the links' semantics (as opposed to exact keyword matching) are generally ignored; only Google allows a user to query the (exact) text contained in hyperlinks [5]. We believe that we will be able to extract crucial semantic material about a Web page from the pages that point to that page. In this article, we will show how to add semantic labels to the edges of the graph of the Web and use them for both characterizing the target nodes' contents and clustering nodes into related groups.

Usually a page does not contain information about itself in its body. This *metainformation* is very useful when answering queries such as looking for a car advertisement, book, picture, etc. While what a page "talks about" can be derived by what is written in the Web page, there will be some added value if we also take into account what *others* think about the page. This is very useful in the case of multimedia content that cannot be otherwise characterized. One only need randomly browse a few Web pages to see that there is a lot of information to be retrieved if the page is used correctly. Let us stress that we will process the text located around links and not only the text that forms the links themselves, unlike what existing systems such as Google do.

Assume a document U is being pointed to by a set of incoming hyperlinks $\{(l_i, s_i)\}$, where s_i are the semantics of link l_i. We claim that the semantics of U are affected significantly by the semantics of the incoming links. Let us consider the following example. If a node U is pointed to by a link, emanating from node S and bearing the semantics "databases", there is a strong indication that the source document S points to the target document U, characterizing it by the term "databases". If there is more than one link from different nodes that point to U bearing the same semantics ("databases"), the indication (as a collective consensus) that U is closely related to the area of databases is stronger – and moreover with high importance.

Semantic hierarchy (ontology) use. Web documents are mainly characterized by extracted keywords and by a rank that takes into account link structures [5]. Finding the similarity between documents is based on exact matching between these terms. This can hardly be called similarity – it is binary matching! For instance, a document d_1 characterized by the keyword list $d_1 = \{snake, desert\}$ would be judged irrelevant to a document d_2 characterized by the list $d_2 = \{adder, Sahara\}$, yet it is arguable that the two lists of keywords (and thus the documents) are in fact related, since an "adder" is a "snake" and the "Sahara" is a "desert". Therefore, d_2 deals with the same concepts as d_1; they are just more specialized. By replacing keywords with concepts and moreover concepts in a hierarchy, a document matching process more flexible than binary matching can be achieved, handling both specializations and generalizations of senses.

THESUS development. We believe that all the above provide a strong motivation for a system that enables the definition and manipulation of thematic subsets of the WWW with rich semantics. In this paper, we present a system to create collections of thematically relevant pages from the WWW and to further distill them into smaller subsets based mainly on their semantic similarity and connectivity features.

The contributions of this paper are summarized in the following:

- A model (THESU) that enables thematic selection of WWW subsets and subsequent enrichment by extracting description from the links pointing to the pages of the subset.
- A mechanism that extracts keywords and enhances hyperlinks with semantics by mapping sets of keywords that describe a Web page to sets of concepts (categories) organized in a hierarchy. In the current implementation, the mechanism employs WordNet [41] as a thesaurus and ontology, the Wu and Palmer similarity measure [42], for computing similarities between terms in a hierarchy.
- A novel similarity measure for weighted sets of terms in a hierarchy that lets one use any sort of proximity-based clustering algorithm (such as DBSCAN [12]).
- A prototype client/server system, called THESUS, that (a) collects URLs and page content based on a given set of keywords, (b) extracts keywords from the collection's incoming and outgoing links (by processing the links' neighboring text in the source URL), (c) maps keywords to categories in the ontology, (d) organizes document collections into subsets each of which contains documents with similar semantics or similar connectivity features, and (e) enables efficient searching on the created collections that focuses on the subsets that match user queries.

The organization of the paper is as follows. Section 3 discusses previous related works. In Sect. 4, we introduce the fundamental concepts and functions on which the THESUS model is based. The architecture of the implemented system follows in Sect. 5, where the operations of the THESUS modules are explained. In Sect. 6, we give some examples and results of the THESUS system that show that if the additional THESUS information is used when querying Web pages, the results are *far* better than without. We conclude in Sect. 7 by summarizing and proposing directions for further work.

3 Related work

3.1 Hyperlink information and semantics

The issue of extracting keywords from links is important in the context of this paper and will be further detailed below. In [30], the idea of "robust" hyperlinks is introduced. Robust hyperlinks are considered to be those that contain descriptive information on the target document. According to the authors, this information can be limited to five words and empirical results are used to prove this. In [6], some experiments on the hyperlink neighboring text result in defining the "anchor window" as being 50 characters long. These experiments count the occurrence of certain keywords in a window of certain width around the hyperlink. Our work capitalizes on these results.

In [40], a system that increases Web searching capabilities is proposed; the system is based on the idea of attaching information to a document concerning its concept and its hyperlinks semantics. The paper proposes a structure for hyperlink information with rich semantics. These semantics emanate from a conceptual hierarchy that varies according to the areas of interest and that can be created by domain experts.

Our working hypothesis is that it is easier to characterize a Web page using information provided on pages that point to it instead of using information that is provided by the page itself. As a result, information can refer to the target of a hyperlink but also to the semantics of the target as they are provided by the source of the hyperlink.

In our system, we rely on the use of a hierarchy of concepts, which can be viewed as a minimalist ontology, and on WordNet in order to compute similarities between words. For more details on ontologies, refer to [16] and for information on WordNet to [41, 1].

3.2 Link analysis and Web document clustering

In most cases, Web document clusters are built based on connectivity between documents [7] (Web structure) and not on semantics that the connectivity might convey. A different approach, such as using Web content mining techniques, mainly performs text mining on the whole document while ignoring the structure of HTML documents and the links between them. It would be very useful to consider both hyperlinks of pages and their contents and then automatically classify large collections of Web documents.

Some works relate to the importance of links as entities that promote semantics in a hypermedia network.

Kleinberg [23] states that "the link structure of a hypermedia environment can be a rich source of information about the content of the environment.... But for the problem of searching in linked environments such as the World Wide Web, it is clear from the prevalent techniques that the information inherent in the links has yet to be fully exploited."

There is a consensus that clustering techniques should be applied to the results of a query rather than to the whole search space in order to discover groups of relevant documents. In [43], a suffix tree clustering (STC) approach is proposed in which the algorithm based on phrases shared between documents is used to create the clusters. In what follows, we detail related work regarding the similarity measures necessary to run the clustering algorithms.

3.3 Similarity measures and document clustering

We are faced with the task of defining a similarity measure among **documents**. In this article, we assume that this boils down to finding the similarity among **sets of weighted terms of a tree** (ontology).

3.3.1 Existing similarity measures between sets

A number of similarities/distances among sets of elements already exist in the literature [14, 18]. The most widely used one, the Jaccard coefficient, is simple: let A and B represent two sets of elements. The similarity between A and B is defined as:

figure=e:/vldb/100/fig1.eps,height=5cm

Fig. 1. Wu and Palmer similarity in a hierarchy

$$\boxed{S_l(a, B) = \frac{|A \cap B|}{|A \cup B|}} \quad \text{Jaccard similarity}$$

In [14], the problem of measuring the similarity or distance between two finite sets of points in a metric space is considered. Some of the distance functions are reviewed, among them the minimum distance link measure, the surjection measure, and the fair surjection measure. All of them take polynomial time algorithms for their computation. In order to be competitive, our measure should also try to be tractable in polynomial time.

In [18], the idea of calculating similarities among sets using the Jaccard coefficient is investigated. The indexing issue for distance/similarity between sets of values is treated in recent work [17], again using the Jaccard coefficient to calculate the similarity. Bidaut et al. in [3] also investigate this with their mediator approach.

The traditional cosine measure from the information retrieval literature (see [35]) has the same behavior as the Jaccard coefficient. Indeed it can be viewed as a direct application of the Jaccard coefficient.

However, in all the aforementioned efforts, the sets of values are considered "flat" (i.e., all values are independent of each other); therefore, only exact matching between values is taken into account by the Jaccard coefficient. With our approach – defining a document in terms of the ontology and calculating the similarity between these sets of terms - documents defined by different sets of words may end up having a very high similarity rating. For instance, recall the example given in the introduction, two documents defined by ("cat", "food") and ("feline", "diet") would have a similarity value different from 0, which is its similarity value according to the Jaccard coefficient. In our case, we want to take into account the proximity that can be derived by the distance between terms in a tree – in our case an ontology.

3.3.2 Similarity between two elements of an ontology

In order to compute the distance between sets of terms of an ontology, we were led to investigate the existing similarity measures in the more simple case of calculating the similarity between two given terms of the ontology. propose different measure inside a taxonomy such as WordNet are proposed in [34, 32, 33], and [25] proposes a comparison between these measures and others, such as Wu and Palmer [42], Miller and Charles, and a novel similarity measure. Desmontils et al. in [11] propose the use of the Wu and Palmer measure in the ontology context. The Wu and Palmer measure is the fastest to compute and is arguably as good as the others ([25]), which is why we chose to use it. We detail its definition and properties in the following paragraph.

Wu and Palmer similarity measure:

Given a tree and two nodes a, b of this tree, we first find their deepest (in terms of depth in the tree) common ancestor c. The similarity measure is computed as follows:

$$S_{W\&P}(a,b) = \frac{2.Depth(c)}{Depth(a) + Depth(b)}$$

In the example shown in Fig. 1, $S_{W\mathcal{I}P}(a,b) = \frac{2\times 2}{4+3} = 0.57$.

3.3.3 Document clustering algorithms

There has been considerable interest in this problem in the information retrieval community [15, 2]. Our problem is more specific since we are clustering Web documents [43] and take *links* into account [23]. Since we are working with categorical data, we need a distance or similarity measure and a density-based algorithm. We have implemented the COBWEB algorithm [15] for categorical data and the DBSCAN algorithm [12, 13] using our similarity measure.

3.3.4 Related systems

Link information is already used by Web search engines to better filter and rank query results, e.g Google's PageRank algorithm prioritizes pages with many incoming or outgoing links. An interesting application that uses hyperlink's structure to group interconnected results is Kartoo (http://www.kartoo.fr). Vivísimo [39] proposes a clustering approach for Web document organization. It makes use of the contents (titles and brief descriptions) that are returned by the underlying search engines. Northern Light [27] classifies each document within an entire source collection into predefined subjects and then, at query time, selects those subjects that best match the search results. Vivísimo does not use predefined subjects; its annotations are created spontaneously. Haveliwala et al. [21] also propose a methodology for evaluating strategies for similarity search on the Web. According to this approach, a Web document u is represented by a set of terms found in the contents, anchor-windows, or links to u. Also, the corresponding weights of this term are used for the Web document description. Thus the Web document u is represented by the following bag:

$$B_u = \{(w_u^1, f_u^1), (w_u^2, f_u^2), \ldots, (w_u^k, f_u^k)\}$$

where w_u^i are terms used in representing u and f_u^i are their corresponding weights.

Then the similarity between documents is measured as the similarity between their bags. The metric used for measuring the similarity of documents is the *Jaccard coefficient*.

A method for classifying and describing Web documents is discussed in [19]. They use inbound links and words surrounding them to describe Web pages. An SVM classifier is trained and used to categorize Web pages. In addition, another method for selecting features and characterizing the classes of Web pages is proposed that uses the expected entropy loss metric. The results of the proposed approach shows that the text in citing documents has a greater descriptive power than the text in the target document itself.

Systems that use a thesaurus to improve results or expand queries are a common idea. There are many articles that deal with this problem, such as [31].

Chekuri et al. [8] propose a system for automatic classification of Web documents into predefined categories using a training set of preclassified documents. The documents are represented using word frequency vectors.

4 THESUS preliminaries

In this section, we introduce the fundamental concepts in the context of the THESUS model that enables thematic selection of WWW subsets and subsequent enrichment by extracting semantics from the links pointing to the pages of the subset. Also, we briefly explain the operations of constructing the ontology and getting the initial set of documents from the Web.

We assume that the WWW is a collection of:

- *pages* uniquely defined by their URL, with each containing a piece of text as its content. Let us underline that pages can be devoid of content (for instance in the case of a picture) yet have other pages pointing to it, with meaningful semantics.
- *links* that connect pages. A link is uniquely defined by its source and target nodes. In our system, links between two pages are perceived as bearers of semantics; thus we are interested in the union of the semantics of all the links pointing from a given page to another given page, and therefore the exact location of the source or target anchor (i.e., the specific location in the source page from which the link emanates) is not of interest.

Link semantics: Assume two pages S and T and the set of links $\{\mathrm{l}_i\}$ that emanate from S and point to T. Also assume a procedure that for each link l_i returns a list of keywords $\{k_j\}$ that characterize the link. When authors of a page want to create a link to another page, they use a small set of keywords to describe the target page. These keywords either appear in the hyperlink source (the text that acts as hyperlink) or in a short area around the hyperlink. In the case that an image is used as the hyperlink source, authors usually use the **alt** attribute to describe the target document. We call this information *link keywords*. Semantics can be defined as the study of meaning in language, and, as a result, semantics refer to the concepts to which extracted keywords map to and therefore assume the existence of an ontology and a mapping mechanism from keywords to concepts. Given an ontology and using Wordnet, we are able to map extracted keywords to concepts of the ontology, so we can easily convert link keywords to what we call *link semantics*.

Ontology: We define ontology as an IS-A tree, but all the results in this paper can be extended to a tree with any sort of "relation" between a parent and child node. Most results are also extendable if the *ontology* is a DAG. We call a word in the ontology a **term** of the ontology. Note that this is a "weaker" definition of an ontology than the ones traditionally proposed in the Semantic Web community.

Document: In this article, we assume that a document d_n is a Web document found at URL_n. It can be abstracted by an identifier (URL_n) and a finite set of weighted terms of the ontology. We note $d_n = \{\mathrm{URL}_n, (w_1, k_1); (w_2, k_2); \ldots\}$.

Ontology creation

In order to provide semantic clustering functions, we need to refer to an ontology of terms that are relevant to our domain of interest. Any hierarchical taxonomy of terms can be used, provided it can be modeled as a tree or DAG. Examples of ontologies that can be used are ontologies suggested by the DARPA Agent Markup Language Program [9] (i.e., ontology on music http://www.daml.org/ontologies/276). For the experiments, we manually created an ontology based on the structure of DMOZ [29] directory. We selected the categories of DMOZ under arts/music, removing duplicates and categories that were not of interest. The resulting ontology is available at http://www.db-net.aueb.gr/thesus/onto/music.rdf. In order for our system to work, we also need to add a one-to-one mapping from each term of the ontology to a set of senses (synset) found in WordNet.

Starting points of crawling

The goal of the document acquisition model is to create a collection of Web documents that possibly relate to the ontology. The acquisition process starts from an initial core of documents and expands it with some of the documents that they point to. This initial core may contain well-known Web directory pages such as the DMOZ, Yahoo, or Google Directory. Focused crawling techniques can be extended to process hyperlink semantics and locate documents with high relevance to the subject.

5 THESUS system architecture

The THESUS system aims at characterizing a set of Web documents using hyperlink information, enhancing its characterization with semantics and distilling the set into thematic subsets (THESUs). The objective is to define a THESU, putting emphasis on the link semantics and the connectivity features among its documents. The developed clustering algorithms further divide the THESU into smaller subsets of pages with similar semantics, thus refining the intention of the initial THESU.

The THESUS system is fully implemented, and a Web demo is available at [36]. The system's components include (see Fig. 2):

- The "document acquisition" module: This module starts from an initial set of URLs D and crawls forward for a user-defined number of times N, following the hyperlinks that carry certain semantics. It selectively collects the documents that are suggested (linked) using certain semantics so that after a maximum of N repetitions an extended set of documents E is created. E contains documents that fall into the same thematic area.
- The "information extraction and enhancement" module: This module extracts keywords from the incoming hyperlinks of the documents of the collection. Then it enhances extracted hyperlink information with semantic information by mapping keywords to concepts in the ontology.
- The "clustering module" that partitions the set E into semantically coherent subsets based either on keywords or on the attached semantic information. The defined clusters are labeled so that the results of the clustering process are more comprehensible and exploitable. Furthermore, the cluster labels enable easier browsing through the document collections.
- The "query module" that: (a) enables searching in the set E, taking advantage of the clusters that are closer to the user's query and ranks the results accordingly; (b) implements link analysis operators, thus allowing the search for thematic authorities, hubs, cocitations, and couplings.

These modules access a relational database in which the document information is stored and employ knowledge from the WordNet 1.6 database [41] and the ontology.

The functions the system provides are: (i) creation of URL collections characterized by certain keywords. This is achieved by combining simple Web searching services provided by search engines (i.e., search for pages that contain specific keywords) and a thematic crawler that expands the search engines' results, (ii) characterization of Web documents or Web subsets at either the keyword or semantics level through the processing of their incoming or outgoing links (by processing a link's neighboring text in the source URL), (iii) clustering of characterized sets of URLs into subsets based both on their characterization or on their intraconnectivity graph.

5.1 Document acquisition

The module uses a Web crawler whose goal is to create a collection of Web documents that possibly relate to the predefined ontology. Let $D = \{d_i\}$ be an initial set of documents related to a certain thematic domain of interest and O be an ontology on the same domain (i.e., Arts, Music, Technology, etc). In the context of this paper, we manually created an ontology containing 176 concepts related to music. The ontology is available at http://www.db-net.aueb.gr/thesus/onto/music.rdf.

The thematic crawler generates the THESUS core set of URLs in two ways, depending on the existence of an initial set of documents D. The concepts $(C_1, C_{11}, C_{12}, \ldots Cm)$ of the ontology are used for the generation of the core in the following manner. Each page in the core set is processed and its hyperlinks, together with a short description from the hyperlink area, are extracted. If the description contains at least one of the ontology terms, then the target of the hyperlink is considered relative to the subject and is added to the set. This procedure recursively collects and analyzes documents and expands the document set. The process stops either after a predefined number of recursions or when a predefined number of documents is collected. The output of the module is a set of pages that relate to the thematic domain described by the ontology.

figure=e:/vldb/100/fig2.eps,width=13.5cm

Fig. 2. THESUS system architecture

5.2 Information extraction and enhancement

5.2.1 Information extraction from hyperlinks

The information extraction module of THESUS uses information from the incoming hyperlinks of a Web document to increase the validity of the extracted description. The use of incoming hyperlinks instead of a document's contents is beyond the scope of this paper; however, below we provide some figures that indicate the quality of results. We encourage the reader to visit the THESUS hyperlink services, which are available online at http://www.db-net.aueb.gr/thesus/services.jsp [36].

In a small-scale experiment on the information extraction module, we selected 50 URLs and obtained their incoming links' characterization using at most 100 incoming links for each URL. We ranked the keywords that appeared in the hyperlinks by the number of occurrences and kept the top ten keywords for each target URL. We presented this description, along with the descriptions provided by Altavista and Google for the same pages, to a group of testers and asked them to rate from 1 to 5 the quality of the description (1 – very bad, 5 – very good), without their knowing which description was which. In more than 50% of the cases, THESUS' descriptions were considered the best of the three, and the average rating for THESUS results was 3.7 out of 5, outscoring the other two systems (Altavista – 1.9, Google – 3.4). It is important to stress that in some of the URLs used in the test, Google and Altavista descriptions either were *provided by human editors* (this is the case for pages in their directory) or contained the title of the page, whereas THESUS' descriptions were *automatically created* and were not based on the contents of the page.

5.2.2 Mapping sets of keywords to sets of concepts

Keyword extraction generates a set of keywords $\{k_j\}$ with the respective weights $\{n_j\}$ for each document d_i. When incoming link information is used, n_j represents the number of hyperlinks pointing to d_i using a particular keyword k_j. Thus each document d_i has the following description: $d_i = (\text{URL}, \{(k_j, n_j)\})$. It is widely accepted that the keyword importance increases proportionally to n_j [5].

THESUS uses WordNet as a means for mapping keywords to concepts. For each keyword we extract and for each concept in the ontology, we can find a set of different senses, given that the keyword is indexed in WordNet. However, not all of the senses provided for an ontology concept are relevant to the subject, nor are all of the senses provided for a keyword relevant to the context of the document. In the former case, the senses of the ontology concepts, which are outside of the ontology scope, are manually rejected by the editor of the ontology. The irrelevant senses of a keyword are rejected automatically through a sense disambiguation process, which is beyond the scope of this paper. In short, the disambiguation process examines the different meanings of a keyword in association with the senses of its context keywords and locates the senses that give the highest similarity score. For example, for the keywords *guitar, flute,* and *wind*, WordNet provides 1, 3, and 8 senses, respectively. The process examines the 24 triplets of senses and gives a score of 0.8 for the triplet (*guitar, flute/transverse flute, wind instrument/wind*) and less than 0.5 to any other combination. Similarly for the keyword set (*storm, cloud, wind*), it gives a score of 0.8 to the triplet of senses (*storm/violent storm, cloud, wind/air moving*) and lower scores to any other combination. These are indications that *wind* has the sense of *wind instrument* in the first set (document) and the sense of *wind as weather phenomenon* in the second set.

Thus we see it is possible to map each keyword k_j to an ontology term t_i with a weight s_j based on the indications given in the previous paragraph. It is common that more than one keyword in $\{k_j\}$ is mapped to the same ontology terms, so the cardinality of $\{t_i\}$ is usually smaller than that of $\{k_j\}$. The weight assigned to each term t_i is computed using the following formula:

$$r_i = \sum_{k_j \to t_i} (n_j \cdot s_j) \Big/ \sum_{k_j \to t_i} n_j$$

Thus each document d_i is represented as $(\text{URL}_i, \{(t_i, r_i)\})$, where $r_i \in [0, 1]$ since $s_j \in [0, 1]$.

5.2.3 Enhanced document

We assume that for a specific domain there exists an ontology O that sufficiently represents the domain semantics. Each set of keywords $\{k_{di}\}$ is mapped to a set of categories $\{c_j\}$ of the ontology O, using a thesaurus, in our case Wordnet. We use a similarity measure, the Wu & Palmer measure [42], to measure the similarity between keywords and categories. The outcome of this process is that every document in the original set is now enriched with:

keywords and weights (indicating the occurrences of a keyword in the incoming links of a page) and
categories of the ontology into which a document is classified and the respective weights.

We use the following notation to define these *enhanced documents*:

Definition: *An enhanced document is the triplet (Doc, K, C), where Doc is the document identifier (i.e., URL), K is the keyword description, and C is the conceptual description of the document.*

To be more precise, K is the set of couples $\{(w_i, k_i)\}$ of weighted keywords that define the document (w_i is a real, k_i is a string). Respectively, C is the set of couples $\{(v_j, c_j)\}$ of weighted concepts that define the document (v_j is a real, c_j is a string. w_i and v_j are not necessarily the same if $i = j$. A weight is a real from the interval [0;1]. A value of 1 indicates total relevance, 0 indicates no relevance. A value of 0.2 would indicate slight relevance.

Remark: these keywords are a positive definition of the document. We do not negatively characterize the document (for instance with words that we would say are definitively not relevant to the document); this remains future work.

5.3 Clustering and labeling module

5.3.1 Clustering

In this phase, the documents are fed into the clustering module. The clustering algorithm is based on a similarity measure between sets of weighted words. We now have the task of finding a relevant definition of a distance between **enhanced documents**. It is important to note that the input of the classifier is a set of documents that have attached to them a set of weighted *terms* of the domain ontology. The goal of our system is the following: *Given documents that are characterized by a (small) set of weighted terms from an ontology, find a way of clustering related documents together*. We propose a clustering scheme based on a novel similarity measure between sets of hierarchically related terms.

Traditionally, in order to achieve this goal, such a user would apply information retrieval techniques such as those described in [35]. However, these techniques most often rely on **exact** keyword matching and do not take into account the fact that the keywords may have some *semantic proximity* between each other. Let us stress that we are running the clustering on the sets of terms that describe the document and that this list can be quite short. For instance, a document might be characterized by the words "cat, food" and another with the word "feline, menu". With traditional exact matching methods, these documents would be judged completely unrelated. However, if we consider an ontology on animals and nutrition, "feline" is a generalization of "cat" and "menu" is a specialization of "food". Given that the ontology has a hierarchical structure, the proximity between "cat" and "feline" and between "food" and "menu" is measured using the Wu & Palmer measure and has a value of (0,1]. We will show that we are able to compute a meaningful similarity measure that takes into account the proximity between document terms in the ontology.

Document similarity measure. This similarity measure will be used both when clustering the documents and when answering queries, but we have yet to optimize the query module of the THESUS system. We expect it will be called upon very often, and thus it is essential that it run as fast as possible, even on large numbers of documents. Therefore, we need to bear in mind the complexity of the calculation of this similarity. For scalability reasons, it must be independent of the number of documents in the database.

Let us not forget that we are calculating similarity between sets of *weighted* words, and not simply words. There has been very little research on creating a similarity measure on sets of elements of a space with a similarity measure defined ([14, 26]). The similarity measure employed in THESUS is a generalization of Wu & Palmer's [42] measure and is defined below. A more detailed discussion on the properties of this measure can be found in [22]. Suffice it to say that the complexity of our similarity measure is quadratic in the number of terms if some values on the ontology are precalculated.

Notations:
Let Ω represent the ontology (set of words in a hierarchy).

We use **cursive capitals** $\mathcal{A}, \mathcal{B}$ to represent sets of weighted words, such as: $\mathcal{A} = \{(w_i, k_i)\}$, with $k_i \in \Omega$ and $w_i \leq 1$.

We note $\mathcal{A} = \{(w_i, k_i)\}$ and $\mathcal{B} = \{(v_i, h_i)\}$

It is in fact an approximation of a more accurate similarity measure that we do not have enough space to detail here, but in most cases it behaves correctly. We define

$$\zeta(\mathcal{AB}) = \frac{1}{2}\left[\left(\frac{1}{K}\sum_{i=1}^{|\mathcal{A}|}\max_{j\in[1,|\mathcal{B}|]}(\lambda_{i,j}S_{W\&P}(k_i,h_j))\right) + \left(\frac{1}{H}\sum_{i=1}^{|\mathcal{B}|}\max_{j\in[1,|\mathcal{A}|]}(\mu_{i,j}S_{W\&P}(h_i,k_j))\right)\right]$$

where $\lambda_{i,j} = \dfrac{w_i + v_j}{2 \times \max(w_i, v_j)}$ and

$$K = \sum_{i=1}^{|\mathcal{A}|}\lambda_{i,x}(i) \quad \text{with} \quad x(i) = x \mid \lambda_{i,x} \times S_{W\&P}(k_i, h_x) = \max_{j\in[1,|\mathcal{B}|]}\lambda_{i,x} \times S_{W\&P}(k_i, h_j))$$

Simply put, K is a normalizing factor that is the sum of all the $\lambda_{i,j}$ that were used.

In a similar way, we can define $\mu_{i,j}$ and H. We refer to [28] for more details on the similarity measure.

Web document clustering algorithm. Clustering aims at organizing patterns into groups, allowing us to discover similarities and differences as well as to derive useful conclusions about them [20]. The module clusters Web documents in order to discover meaningful groups. The problem is considerably different from the case of points in a metric space. In our case, the objects to be clustered are sets of (weighted) terms of a domain ontology that correspond to categories of a domain ontology.

In this space, there are no coordinates and ordering as in a Euclidean metric space. We can only compute the similarity between documents given an appropriate similarity measure between sets of weighted categories. We embedded the similarity measure in two clustering algorithms, DBSCAN and COBWEB, that we used for evaluating the behavior of the similarity measure. Our objective here is to cluster Web documents in order to discover meaningful groups of pages. The classic clustering problem is the case of points in a metric space. In our case, the objects to be clustered are sets of (weighted) terms of a domain ontology between which we have created a similarity measure.

In this space, there are no coordinates and ordering as in a Euclidean metric space. We can only compute the similarity between documents given an appropriate similarity measure between sets of terms. We embedded the similarity measure in two clustering algorithms, DBSCAN and COBWEB. Let us stress that in DB-SCAN's case, the original algorithm was designed for geographical databases. By defining a semantic similarity measure, we are able to apply the algorithm with little modification, and with good results.

The output of the module is the **partitioned set of documents**, some of which are considered as noise (in the case of DBSCAN).

5.3.2 Cluster labeling

Once the clusters have been found, a very important issue is their labeling (i.e., the assignment of a succinct yet descriptive set of categories to each cluster in order to facilitate user navigation and querying). Grouping documents together is itself a semantic enhancement. We would also like to find appropriate labels for each cluster for the following reasons:

- Simply grouping documents together does not provide a means of characterizing this set.
- We need some way of calculating which cluster a given query is closest to. This would in particular reduce the task of finding that cluster to which we should apply a keyword query to a simple similarity measure between that query (i.e., the set of keywords) and all the labels of the clusters.
- Giving a more precise characterization to the cluster will enable easier browsing through the set of documents as a whole.

Labeling scheme. We want these labels to have some sort of significance. The labeling process is summarized as follows:

- Construct U, the union of all concepts that appear in at least one document of the cluster.
- For every concept C_i in U, calculate either the number of documents in the cluster that it appears in or the percentage of documents of the cluster that are characterized by a given concept.
- Reduce the number of concepts in U without losing information, using the ontology. In this step, less important categories (with low percentage or very specific ones) are replaced by their superseding categories and then similar categories are grouped together. This replacement reduces the number of concepts in the label to the desired one.

6 Testing and experiments

6.1 Experimental setup

This section presents the results of the experiments performed with THESUS on three different Web document sets that show the final results of the document clustering scheme. The experimental scenarios follow:

We use documents that have been described and organized into categories by humans. For this purpose, we use the DMOZ Web directory and compare the results of our clustering to the initial categories of DMOZ. We assume that the main factors that affect the clustering results are:

- the quality of the keyword characterizations for each document
- the method of estimating similarity between documents
- the proximity of the initial categories

The experiments aim to provide a comparison on how these factors affect the clustering quality. The process of creating, characterizing, and clustering the document collections is performed as follows:

1. We select documents from various DMOZ categories that belong to different levels of the DMOZ hierarchy.
2. We characterize the selected documents using keywords extracted from the descriptions provided by the DMOZ editors, from the documents' contents, or from the incoming links of these documents.
3. We enhance the document characterization by mapping the extracted keywords to the respective concepts of our thematic hierarchy.
4. We compute the similarity between documents using either cosine similarity on the keywords extracted in step 2 or the THESUS similarity measure (THESIM) on the respective concepts

Document clustering evaluation

The quality of the clustering results is measured with the use of two methods, **F-measure** [24] and **Rand Statistics** [37]. These are both external quality measures, which implies that they are defined to measure the degree of correspondence between a predefined categorization of a document set, D, and the clustering that results after the application of a clustering algorithm to D.

The three document sets

Similar experiments that cluster Web document sets usually evaluate the clustering quality using external quality measures and are compared to flat classifications of documents such as the documents of the TREC Web track [38]. Even when a multilevel classification is employed (such as Yahoo or DMOZ), documents are selected from sibling categories either from the top level (i.e., sports, business) or from lower levels (i.e., soccer, volleyball) [10]. These categories usually contain a few URLs and several subcategories with their URLs. Experiments usually flatten the top categories by including all the contents of their subcategories. They compare the final clustering scheme with respect to the selected top categories.

In our experiments, we adopt the same method. We take the URLs that fall under the arts/music branch of DMOZ excluding the "bands and artists" subcategory, and all "by-letter" subcategories, that mainly contain surnames, company names, etc., with no conceptual content. This results in approximately 30,000 URLs from 2155 different categories. This is the first document categorization we compare and that we name (FULL-SET).

We should stress here that the ontology used in this experiment is not identical to the part of the DMOZ tree from which we retrieved the test document set. This is done because terms in DMOZ paths do not necessarily appear in WordNet; thus we cannot map extracted keywords to them and moreover we judge that the DMOZ topic hierarchy does not completely express a hierarchy of concepts related to music. For example, when DMOZ distinguishes between

Table 1. Number of documents per category in the LEVEL1 document set

Sound files	1933
Instruments	6082
Lyrics	853
Vocal	733
Marching	928
Styles	13061

Table 2. Number of documents per category in the LEVEL2 document set

Electronic	378
Keyboard	784
Percussion	293
Squeezebox	94
Strings	2627
Winds	1735

"Gamelan" and "Indian" music, whereas in our ontology only the term "World Music" exists, it is expected that documents from both categories of DMOZ will be grouped together in our approach. Thus it is not expected that the clusters in our approach will be exactly identical to the groups of documents classified to the DMOZ paths.

We create two more document sets with flattened categories. The first set (LEVEL1) contains the most populated subcategories of arts/music. The categories contain approximately 25,000 documents and are listed in the following table (Table 1).

The second set (LEVEL2) contains approximately 6000 documents under the path arts/music/instruments. Categories are flattened again. The number of documents for each of the categories is listed in the following table (Table 2).

It is straightforward that
LEVEL2$\subset$LEVEL1$\subset$FULL-SET.

A comparison of document characterization techniques

In order to find the best way to describe a Web document, we produced three different characterizations for each document in all the sets. The characterizations contained:

- the keywords given by DMOZ editors (10 to 20 keywords), the number of times that each keyword appears in the description (DMOZ)
- the ten most frequent keywords extracted from the document's contents, the number of appearances (CONTENT)
- the keywords that appear in the 100 incoming links (at most), the number of incoming links that each keyword appears in (INLINKS). It is interesting to note that the mean number of keywords extracted in this way is less than 10.

Different keywords that appear in a document description may be mapped to the same ontology concept, so the concepts that constitute the semantic description of documents are usually less than the respective keywords.

Comparing the cosine similarity measure to our measure

In order to compare the effectiveness of our method for defining document similarity using a hierarchy of terms, we map the descriptions extracted in the previous step to descriptions that use terms of the hierarchy and cluster documents using THESUS similarity measure (THESIM) to compute document similarity. We cluster documents again using the keyword descriptions and the cosine similarity measure (COSINE).

Evaluating clustering quality using two external quality measures

In the experiments we performed, we assigned different values to the input parameters of the DBSCAN algorithm. We evaluate clustering quality each time, using Rand Statistic and F-measure. The highest values for both measures are depicted in the results table (Table 3).

The number of clusters produced in an experiment may differ significantly depending on the input parameters of the clustering algorithm. For each document set, description, and similarity measure combination, we repeat the clustering process several times with different input parameters and compare the produced clustering scheme with the original categorization (2155 categories for the FULL-SET, 6 categories for LEVEL 1 and LEVEL 2). We keep the clustering scheme with the maximum likeness to the original categorization in respect to the F-measure and Rand Statistics values.

The two measures compute the covering between the produced clustering of a document set and a predefined categorization of the same set, which is considered as the baseline of comparison. In our case, the baseline is the categorization of the documents by DMOZ's editors. In general, external quality measures for clustering, such as F-measure and Rand Statistics, examine all the possible pairs of documents in a document set.

More specifically, the former validity index evaluates the effectiveness of a clustering based on the recall and precision of the defined clusters with respect to the predefined set of categories. *Recall* measures the proportion of the documents in a category that are clustered together, while *precision* is the proportion of documents clustered together that belong to the same initial category. On the other hand, *Rand Statistics* measures the proportion of the total number of documents pairs that both clustered together and belong to the same category or belong to different clusters and different categories. In general terms, the quality of defined clustering with respect to a predefined categorization increases when pairs of documents belonging to the same category are clustered together and decreases when documents from the same category are assigned to different clusters.

As shown in Table 3, the use of our similarity measure on the sets of concepts outperforms this of cosine similarity on the sets of keywords. The results achieved using incoming link descriptions are better than those achieved using most frequent keywords in the documents' content and comparable to descriptions provided by humans. It is also evident that the computation of THESIM is not computationally expensive. The transition from keywords to concepts of an ontology results in significantly smaller description vectors since more

Table 3. Clustering results for the three datasets with optimal input parameters

		FULLSET THESIM	FULLSET COSINE	LEVEL1 THESIM	LEVEL1 COSINE	LEVEL2 THESIM	LEVEL2 COSINE
DMOZ	Input						
	MinDoc	1	1	5	5	5	5
	MinSim	1	0.4	0.7	0.2	0.65	0.2
	Output						
	Clusters	1120	2002	7	17	6	19
	Clust.doc.	16436	17921	14315	22967	2853	5577
	Tot. doc.	22575	29115	18252	23590	4615	5911
	F-measure	0.182	0.196	0.724	0.533	0.694	0.589
	Rand Stat	0.913	0.849	0.502	0.482	0.513	0.378
	Cl. time	80	90	40	60	0.5	0.5
	Init. Time	689	850	499	2285	25	78
CONTENT	Input						
	MinDoc	1	1	1	1	10	25
	MinSim	1	0.45	0.7	0.15	0.7	0.45
	Output						
	Clusters	598	1258	6	8	8	6
	Clust.doc.	9891	13507	10006	17545	1679	1663
	Tot. doc.	12774	21904	10242	17743	3011	4636
	F-measure	0.162	0.160	0.720	0.492	0.623	0.483
	Rand Stat	0.917	0.849	0.513	0.501	0.518	0.485
	Cl. time	52	60	40	60	0.60	0.6
	Init. Time	309	650	131	562	12	38
INLINKS	Input						
	MinDoc	1	1	5	5	5	5
	MinSim	1	0.4	0.65	0.1	0.65	0.1
	Output						
	Clusters	879	928	4	23	5	5
	Clust. doc.	12080	14397	11312	13534	2737	4062
	Tot. doc.	17913	18944	14248	15048	3473	4147
	F-measure	0.200	0.130	0.730	0.592	0.639	0.511
	Rand Stat	0.751	0.926	0.500	0.441	0.503	0.330
	Cl. time	88	93	52	60	2.7	3.1
	Init. time	365	565	580	598	34	385

Input: Input parameters, MinSim is the minimum similarity between two documents in a cluster, MinDoc+1 is the minimum number of documents in a cluster.
Output: The number of clusters produced and the number of documents clustered. The remaining documents are considered as noise.
F-measure, Rand Stat.: The two clustering quality measures.
Init. time: The time (in seconds) needed to calculate the distance between all pairs of documents.
Cl. time: Average time (in seconds) for clustering the set of documents

than one keyword is mapped to the same concepts or to no concept at all. The complexity of THESIM with conceptual descriptions is $O(n * p)$, where n and p are the cardinality of the sets of concepts that describe each document. It is interesting to note that both n and p are very small (~ 10 terms). The complexity of the equivalent COSINE measure would be $O(k)$, where k is the number of different keywords that appear in the document descriptions, which can be very large.

A closer look at the figures in Table 3 shows that the number of total documents for the three different description techniques differs significantly. This is because for each technique the source of information is not available for all the documents in the set. For example, DMOZ provides descriptions for 22,575 documents of the FULL-SET, whereas when we tried to access the documents and parse their contents, we only got information for 12,774 of them. The main reason for this is that either the documents were not accessible at the moment of the experiment or their contents could not be automatically parsed. On the other hand, hyperlink information was more available, so we got information from incoming hyperlinks for 17,913 of the documents in the set. The same variation occurs for the two other document sets (LEVEL 1, LEVEL 2).

Additionally, when the descriptions that contain extracted keywords are replaced with descriptions containing concepts from the ontology and the THESIM is used instead of the

Table 4. Clustering in THESUS using keyword and concept descriptions

	KEYWORDS	CONCEPTS
Input		
MinDoc	1	5
MinSim	0.45	0.65
Output		
Clusters	6	5
Clust. doc.	1382	2737
Tot. doc.	4147	3473
F-measure	0.519	0.639
Rand Stat	0.45	0.503
Cl. time	3.1	2.7
Init. Time	1407	34

cosine similarity, the number of documents to be clustered decreases. The reason for this is that not all keyword descriptions are relevant to the domain of interest and as a matter of fact they are not mapped to ontology concepts. This illustrates the ability of the mapping mechanism to discard the irrelevant documents' descriptions before clustering the documents.

6.2 Comparing keyword and conceptual descriptions

The quality of results presented in Table 3 is influenced by two factors: the similarity measure used in each case and the quality of mapping keywords to concepts. When using concepts instead of keyword descriptions for the documents, the similarity given by THESUS is higher than the cosine similarity. Keywords that are identical in the two documents map to the same concept, whereas keywords that are not identical but bear similar meanings may be mapped to the same concept, thus increasing the similarity between the two documents. In order to get an indication on the influence of the mapping process in the quality of clustering, we performed an additional experiment in which we used the dataset of LEVEL 2, with keyword descriptions extracted from the pages' content and the respective concepts. In this experiment, we use THESIM for sets of keywords, employing WordNet instead of the domain ontology as a hierarchy. In this case, the computation of similarity between documents is much slower since WordNet must be accessed for every keyword and preprocessing is not feasible. The clustering process is also much longer to complete. However, clustering quality is slightly better when keywords are used. The results are presented in Table 4.

6.3 Complexity analysis

Apart from the quality of the clustering, the efficiency and scalability of the system is important. The time required for clustering a set of Web documents is influenced by many factors such as the number of documents, the number of concepts that describe each document, and the complexity of the clustering algorithm. The total time needed for clustering the set is also affected by the system parameters such as CPU speed and the available amount of main memory.

We assume that N is the number of documents in the set and M is the mean number of concepts for each document in the set. We also assume that the similarities between all pairs of concepts in the ontology have been precalculated and stored in main memory. The similarity between two documents using our measure is calculable in $O(M^2)$, assuming that the time to access the precalculated Wu & Palmer similarity for two concepts in the ontology is $O(1)$.

In [12], DBSCAN is used in the context of spatial databases and assumes that the elements to be clustered are points in a metric space of a given dimension, usually 2 or 3. The distance measure used is a simple Euclidian distance. In order to compute the neighborhoods of a document, the points are inserted into an R*-Tree [4]. In our case, we are not in a metric space with known dimension. We cannot use an R*-Tree. Instead, we simply precalculate the similarity between all the N documents in the set. We save them in N different lists of size N. This costs us $O(N^2)$. Then we sort each list, using Quicksort. This well known algorithm has an average complexity of $O(N \log N)$ to sort one list of length N. To sort our N lists, we have a complexity $O(N^2 \log N)$. Once this preprocessing phase is complete, we can apply the algorithm.

The time complexity of the clustering procedure is based on the average complexity of defining density-connected sets of documents, i.e., identifying the neighbors of the documents in the database. In our system, since the similarity of a document d_i with all other documents is precalculated in an ordered list, the time complexity of defining the list of the documents closest to it (i.e., with $\zeta > MinSim$) using a dichotomic method is $O(\log N)$. The algorithm does this task once for every document in the set; thus the complexity is $O(N \log N)$.

7 Conclusions

Web search engines answer user queries with a set of pages that match the majority of terms in the query. Information retrieval algorithms rank the results based on the distance of the query to the document contents. Yet document contents have been created by the document authors, who in some cases are interested in getting a high rank for their pages. Such algorithms can be manipulated by page authors by adding text that can affect search engines' ranking for a query. With hyperlink information, the introduction of such bias becomes harder. Searching in the WWW is a task of very high importance, in social and financial terms, as hundreds of millions of users worldwide, with diverse profiles, are searching for pages relevant to their requests. Currently this task is carried out mainly by submitting queries to search engines. The search criteria are based on the pages' contents, ignoring the additional semantics emanating from links, a cornerstone entity of the WWW.

In this paper, we capitalize on this observation and present the architecture of a system, THESUS, that collects Web documents of a thematic domain and extracts information of the collection's incoming and outgoing links (by processing a link's neighboring text in the source URL). The system enhances extracted information with semantics using an ontology and a thesaurus and populates a relational database with all this information. A clustering module is able to detect subsets of the initial document set that have similar semantics,

assigned by incoming links. Also, we discuss some experimental results using the implemented system.

Current and future work addresses many parts of the system:

- The manually created hierarchy of concepts will be semi-automatically connected to WordNet
- We are developing an interface that for each concept in the hierarchy suggests the most possible WordNet senses (using sense disambiguation techniques) and allows the creators of the ontology to clarify the meaning of each concept.
- The thematic crawler is becoming adaptive. In each crawl, it extends its vocabulary by adding keywords that coappear frequently with keywords from the ontology.
- The query system can be made more efficient. This involves optimizing the mapping techniques that we use in order to determine which word of the ontology a generic word should be mapped to. This is a topic that we are currently working on and that is very closely linked to this article since it is used to define the small sets of terms that define a Web document.

Acknowledgements. We would like to thank the following people for their helpful discussions on various topics related to this paper: G. Cobena and S. Abiteboul (general work on the SPIN projet, a similar effort to construct a personal thematic warehouse). Ch. Froidevaux, C. Nicaud, K. Nørvåg, B. Safar, and L. Segoufin (similarity measures and distances). J.P. Sirot (ontologies). P. Rigaux and P. Veltri (R*-Trees).

References

1. Al-Halami R, Berwick R et al (1998) In: Fellbaum C (ed) WordNet, an electronic lexical database. Bradford Books, location of publisher
2. Aggarwal C, Gates S, Yu P (1999) On the merits of building categorization systems by supervised clustering. In: Proceedings of the 5th ACM-SIGKDD, location, day month 1999, pp 352–356
3. Bidault A, Safar B, Froidevaux Ch (2002) Proximit'e entre requetes dans un contexte mediateur. RFIA-2002, pp 653–662
4. Beckmann N, Kriegel HP, Schneider R, Seeger B (1990) The R*-Tree: an efficient and robust access method for points and rectangles. SIGMOD Conference 1990
5. Brin S, Page L (1998) The anatomy of a large-scale hypertextual Web search engine. In: Proceedings of the 7th international World Wide Web conference, Brisbane, Australia, April 1998, page range
6. Chakrabarti S, Dom B, Gibson D, Keinberg J, Raghavan P, Rajagopalan S (1998) Automatic resource list compilation by analyzing hyperlink structure and associated text. In: Proceedings of the 7th international World Wide Web conference, location, month 1998, page range
7. Chakrabati S, Dom B, Gibson D, Kleinberg J, Kumar S, Raghavan P, Rajagopalan S, Tomkins A (1999) Mining the link structure of the World Wide Web. IEEE Comput 32(8):60–67
8. Chekuri C, Goldwasser M, Raghavan P, Upfal E (1997) Web search using automatic classification. In: Proceedings of the 6th international World Wide Web conference, location, month 1997, page range
9. The DARPA Agent Markup Language Ontology Library. http://www.daml.org/ontologies/
10. Dumais S, Chen H (2000) Hierarchical classification of Web content. In: Proceedings of the 23rd ACM international conference on research and development in information retrieval, location, month 2000, page range
11. Desmontils E, Jacquin C (2002) Indexing a Web site with a terminology oriented ontology. In: Cruz IF, Decker S, Euzenat J, McGuinness DL (eds) The emerging semantic Web. IOS Press, Amsterdam, pp 181–198
12. Ester M, Kriegel HP, Sander J, Xu X (1996) A density based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the 2nd international conference on knowledge discovery and data mining ACM-SIGKDD, location, month 1996, page range
13. Ester M, Kriegel HP, Sander J, Wimmer M, Xu X (1998) Incremental clustering for mining in a data warehousing environment. In: Proceedings of the 24th VLDB conference, location, month 1998, page range
14. Eiter T, Mannila H (1997) Distance measures for point sets and their computation. Acta Informat 34: 1997
15. Fisher D (1987) Knowledge acquisition via incremental conceptual clustering. Mach Learn 2:139–172
16. Guarino N (1998) Formal ontology and information systems. In: Guarino N (ed) Formal ontology in information systems. In: Procedings of the 1st international conference, Trento, Italy, month 1998. IOS Press, Amsterdam
17. Gionis A, Gunopulos D, Koudras N (2001) Efficient and tunable similar set retrieval. ACM-SIGMOD , location, month 2001, page range
18. Green J, Horne N, Orlowska E, Siemens P (1996) A rough set model of information retrieval. Theor Informat 28:273–296
19. Glover EJ, Tsioutsiouliklis K, Lawrence S, Pennock DM, Flake GW (2002) Using Web structure for classifying and describing Web pages. In: Proceedings of the WWW conference , location, month 2002, page range
20. Halkidi M, Batistakis Y, Vazirgiannis M (2001) On clustering validation techniques. Intell Inform Sys J volume number:page range
21. Haveliwala TH, Gionis A, Klein D, Indyk P (2002) Evaluating strategies for similarity on the Web. In: Proceedings of the WWW Conference, location, month 2002, page range
22. Halkidi M, Nguyen B, Varlamis I, Vazirgiannis M (date) THESUS: Organizing web document collections based on semantics & clustering. Technical Report. (available at http://www.db-net.aueb.gr/pubs.php)
23. Kleinberg J (1999) Authoritative sources in a hyperlinked environment. J ACM 46:page range
24. Larsen B, Aone C (1999) Fast and effective text mining using linear-time document clustering. In: Proceedings of KDD-99, San Diego, month 1999, page range
25. Lin D (1998) An information-theoretic definition of similarity. In: Proceedings of 15th international conference on machine learning, location, month 1998, page range
26. Niiniluoto I (1987) Truthlikeness. Reidel, Dordrecht
27. The Northern Light search engine: http://www.northernlight.com
28. Nguyen B, Varlamis I, Halkidi M, Vazirgianis M (2002) Clustering Web documents using an ontology. Technical Report Verso
29. ODP – Open Directory Project, http://dmoz.org/
30. Phelps T, Wilensky R (2000) Robust hyperlinks cost just five words each. UC Berkeley Computer Science Technical Report UCB//CSD-00-1091. Berkeley, CA

31. Qui Y, Frei HP (1994) Improving the retrieval effectiveness by using a similarity thesaurus.
32. Resnik P (1995) Using information content to evaluate semantic similarity in a taxonomy. IJCAI-95, pp 448–453
33. Resnik P (1999) Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language. J Artif Intell Res volume number:page range
34. Richardson R, Smeaton A, Murphy J (1994) Using WordNet as a knowledge base for measuring semantic similarity between words. In: Proceedings of the AICS conference, Dublin, month 1994, page range
35. Salton G, McGill M (1983) Introduction to modern information retrieval. McGraw-Hill, New-York
36. Thesus Web page: http://www.db-net.aueb.gr/thesus/
37. Theodoridis S, Koutroubas K (1999) Pattern recognition. Academic Press, New York
38. Web research collections – TREC Web Track. http://www.ted.cmis.csiro.au/TRECWeb/
39. Vivisimo search engine: http://www.vivisimo.com/
40. Varlamis I, Vazirgiannis M (2001) Web document searching using enhanced hyperlink semantics based on XML. In: Proceedings of IDEAS 2001 conference, location, month 2001, pp 34–43
41. Wordnet Web site: http://www.cogsci.princeton.edu/∼wn/
42. Wu Z, Palmer M (year) Verb semantics and lexical selection. In: Proceedings of the 32nd annual meetings of the associations for computational linguistics, location, month year, pp 133–138
43. Zamir, E (1998) Web document clustering: a feasibility demonstration. In: Proceedings of ACM-SIGIR ‘98, location, month 1998, page range