Automated Subject Classification of Textual Documents in the Context of Web-Based Hierarchical Browsing

Koraljka Golub

Koraljka Golub obtained her Ph.D. from Lund University, Sweden, on the topic presented in this article. She currently works as a research officer at UKOLN, University of Bath, United Kingdom, focusing on knowledge organization systems. Two of her recently completed projects are EnTag, on the potential of combining social tagging with Dewey Decimal Classification and Library of Congress Subject Headings, and TRSS, a scoping study of terminology registries for UK furhter and higher education needs. The main ongoing project in which she is involved is EASTER, on evaluation methodology and algorithms for automated classification and indexing. Her home page is available at http://www.ukoln.ac.uk/ukoln/staff/k.golub/.

Abstract. While automated methods for information organization have been around for several decades now, exponential growth of the World Wide Web has put them into the forefront of research in different communities, within which several approaches could be identified: 1) machine learning (algorithms that allow computers to improve their performance based on learning from pre-existing data); document clustering (algorithms for unsupervised document organization and automated topic extraction); and, string matching (algorithms that match given strings within larger text). Here the aim was to automatically organize textual documents into hierarchical structures for subject browsing. The string-matching approach was tested using a controlled vocabulary (containing pre-selected and pre-defined authorized terms, each corresponding to only one concept). The results imply that an appropriate controlled vocabulary, with a sufficient number of entry terms designating classes, could in itself be a solution for automated classification. Then, if the same controlled vocabulary had an appropriate hierarchical structure, it would at the same time provide a good browsing structure for the collection of automatically classified documents.

1 Introduction

Automated subject classification research began with the availability of electronic text in the early 1950s and has been a challenging topic ever since. Interest especially grew in the 1990s when the number of digital documents started to increase exponentially. Because of the high human costs of manual subject classification and the ever-increasing amount of available documents, there is a danger that the established objectives of bibliographic systems could be lost sight of (Svenonius 2000, 20-21). Instead, automated means could be a solution to preserve them (ibid., 30). Apart from bibliographic systems, automated subject classification of textual documents is used today in a wide variety of applications. For example, hierarchical organization of documents is used for browsing, focused crawling, e-mail filtering and many other such applications (see Sebastiani 2002, 6-9).

One could argue that the most frequent approach to automated classification is machine learning (for a thorough review see Sebastiani 2002), which requires training documents from which to "learn", and consequently performs well if new documents are similar enough to the ones used for training. Document clustering (see Jain et al. 1999 for a general clustering overview) is another approach; it does not require training documents but instead compares the documents to be classified to one another, resulting in clusters of documents that are found to be similar using the algorithm. The third and least common approach reuses the intellectual work invested in creating a controlled vocabulary and applies string matching against the controlled vocabulary (for examples see Toth 2002). A great deal of research in automated classification focuses on improving algorithm performance *per se*. If application context exists it is commonly subject searching as opposed to hierarchical browsing – in spite of the fact that organizing web pages into hierarchical structures for subject browsing has been gaining recognition as an important tool supporting information seeking (Large et al. 1999, 192; Koch and Zettergren 1999). Moreover, a combination of automated subject classification in the context of browsing has hardly been studied at all, which makes this work a particularly relevant contribution to current automated classification research.

This paper is an overview of results obtained during four years of PhD research, parts of which have been individually published and will be referred to throughout the paper (Golub and Ardö 2005; Golub 2006a; Golub 2006b; Golub 2006c; Golub et al. 2006; Koch et al. 2006; Golub et al. 2007; Golub and Lykke Nielsen 2009). The aim of the work was to study approaches to automated subject classification in the context of hierarchical browsing. One major focus was to explore a string-matching approach to automated subject classification that does not require pre-classified documents but instead makes use of the intellectual work that was put into building a good controlled vocabulary. The advantages and challenges of automatically classifying web pages were examined in particular detail. Automated subject classification was examined not only by comparison with pre-assigned classes, which is the prevalent evaluation method, but also through users' judgements on the correct placement of documents while browsing. Browsing behaviour was studied in two different environments: a

large manually classified web page collection, and a collection of automatically classified web pages.

The principle research questions were as follows:

1) How is hierarchical browsing in a large web service used (if at all)?
2) Are established classification schemes such as the Dewey Decimal Classification (hereafter DDC) (OCLC 2010) and the Engineering Information thesaurus and classification scheme (hereafter Ei) (Milstead 1995) suitable for hierarchical Web-based browsing?
3) Which approaches to automated subject classification are in use, and what are their advantages and disadvantages, especially in relation to hierarchical Web-based browsing?
4) What are the challenges of applying a string-matching classification algorithm to a collection of pre-classified Web pages?
5) How can the performance of automated subject classification using the string-matching algorithm be improved?
6) What level of performance can the string-matching algorithm yield when applied to a collection of pre-classified paper abstracts and evaluated by comparison to pre-assigned classes?
7) What level of performance can the string-matching algorithm yield when applied to a collection of harvested Web pages and evaluated by end-users?

The paper is structured as follows: the second section (Background) provides general information such as definitions and research challenges; the third section (Methodology) describes the principle classification algorithm, document collections and performance measures; in the fourth section (Results), the principle results are presented and discussed; finally, concluding remarks and the implications for further research are presented (Concluding remarks).

2 Background

2.1 Terminology

Classification is, for the purpose of this paper, defined as "…the multistage process of deciding on a property or characteristic of interest, distinguishing things or objects that possess that property from those which lack it, and grouping things or objects that have the property or characteristic in common into a class. Other essential aspects of classification are establishing relationships among classes and making distinctions within classes to arrive at subclasses and finer divisions." (Chan 1994, 259).

Automated subject classification (hereafter referred to simply as automated classification) denotes machine-based organization of related information objects into topically related groups. In this process, human intellectual processes are replaced by, for example, statistical and computational linguistics techniques. In related literature, automated classification can also be referred to as *automated indexing* (Moens 2000; Lancaster 2003), and the terms *automatic* and *automated* are both used. Here the term *automated* is chosen because it more directly implies that the process is machine-based.

In difference to searching, *browsing* in general relies on recognition of patterns (e.g. sequences of words) rather than recall of search terms from memory (Large et al. 1999, 179). *Hierarchical browsing* in this paper refers to using a hierarchical tree structure in which information resources are organized by topic.

2.2 Approaches to automated classification

As discussed in Golub (2006a), three major approaches to automated classification of text can be distinguished, which are viewed in this work in the specific context of hierarchical subject browsing: machine learning, document clustering, and string matching (research question **3**).

There are considerable terminological inconsistencies in related literature, and the terminology used here is further explained below.

In document clustering, both clusters (classes) into which documents are classified and, to a limited degree, relationships between them, are produced automatically. Labelling the clusters is a major research problem, with relationships between them, such as those of equivalence, related-term and hierarchical relationships, being even more difficult to derive automatically (Svenonius 2000, 168). In addition, "…[a]utomatically-derived structures often result in heterogeneous criteria for category membership and can be difficult to understand." (Chen and Dumais 2000, 146). Also, the labels of clusters and the relationships between them change as new documents are added to the collection; unstable class names and relationships are user-unfriendly in information retrieval systems, especially when used for subject browsing.

Machine learning is the most widespread approach to automated classification of text. (An alternative term within the community is 'text categorization'.) Here, the characteristics of subject classes, into which documents are to be classified, are learnt from documents with pre-existing manually assigned classes. However, manually classified documents are often unavailable, in many subject areas, for different document types or for different user groups. If one were to judge by the standard Reuters Corpus Volume 1 collection (Lewis et al. 2004), some 8,000 training and testing documents would be needed per class. A related problem is that text categorization algorithms perform well on new documents only if they are similar enough to the training documents. The problem of document collections was pointed out by Yang (1999), who showed how even slight differences between versions similar versions of the same document collection had a strong impact on performance.

Traditionally, research in machine learning seems to be focused on improving algorithm performance, and experiments are conducted under laboratory-like conditions. Also, studies in which web pages are categorized into hierarchical structures for browsing do not generally involve well-developed classification schemes, but home-grown structures such as search engines' directories that are not well structured and/or maintained. Moreover, often only a few categories with one or two hierarchical levels are used in experiments, so each consequently contains an 'unbrowseable' number of documents.

In string matching, text from the document to be classified is compared with controlled vocabulary terms representing classes, and then, following a set of more or less heuristic rules the document is assigned (some or all of) the matched classes. A major advantage of this approach is that it does not require training documents, while still maintaining a pre-defined structure. If using a well-developed classification scheme, it is also suitable for subject browsing in information retrieval systems. This would be less true with automatically created classes and document clustering structures or with home-grown directories that were not created in compliance with professional principles and standards. Apart from improved information retrieval, another motivation to apply controlled vocabularies in automated classification is to reuse the valuable work that has gone into creating such a controlled vocabulary (see also Svenonius 1997).

2.3 Evaluation challenge

According to the ISO standard on methods for examining documents, determining their subjects, and selecting index terms (International Organization for Standardization 1985), manual subject indexing is a process involving three steps: 1) determining the subject content of a document, 2) a conceptual analysis to decide which aspects of the content should be represented, and 3) translation of those concepts or aspects into a controlled vocabulary. These steps are based on a specific policy with respect to the document collection and target user groups, for example in terms of exhaustivity (the number of concepts to index) and specificity (the depth of detail to index). Thus, when evaluating automatically assigned classes against manually assigned ones, it is important to know the collection's indexing policies.

Another problem to consider when evaluating automated classification is that certain subjects in document collections are erroneously assigned. When indexing, people make errors such as those related to the exhaustivity policy (too many or too few subjects become assigned), specificity of indexing (which usually means that the assigned subject is not the most specific one available): they may omit important subjects, or assign an obviously incorrect subject (Lancaster 2003, 86-87).

Moreover, it has been reported that different people, whether users or professional subject indexers, would assign different subjects to the same document. Studies on inter- and intra-indexer consistency report generally low consistency between indexers (Olson and Boll 2001, 99-101). Markey (1984) reviewed 57 indexer consistency studies and reported that consistency levels ranged from 4% to 84%, with only 18 studies showing over 50% consistency. There are two main factors that seem to affect it:

1) Higher exhaustivity and specificity of subject indexing both lead to lower consistency, i.e. indexers choose the same first term or class notation for the major subject of the document, but the consistency decreases as they choose more subjects;
2) The bigger the vocabulary, or, the more choices the indexers have, the less likely it is that they will choose the same terms or class notations (Olson and Boll 2001, 99-101).

Complementing the above, a number of issues have been discussed in the literature, such as what is the chief source of evidence in document interpretation (the document, the user, the domain or the request), or what constitutes a valid process of indexing (what processes does the subject analyst go through when indexing and when is one finished with the process) (Tennis 2009). "The phenomenon of indexing is complex. Our theories of document interpretation have showed us just a few of the factors that influence our understanding of the act and its contingencies" (ibid., 198).

Because of the above, without a thorough qualitative analysis of automatically assigned classes, one cannot be sure whether, for example, the classes assigned by the algorithm, but which are not manually assigned, are actually wrong, or if they were left out by mistake or because of the indexing policy. Today, however, evaluation in automated classification experiments is mostly conducted under controlled conditions, ignoring these factors. As Sebastiani (2002, 32) puts it, "…the evaluation of document classifiers is typically conducted experimentally, rather than analytically. The reason is that… we would need a formal specification of the problem that the system is trying to solve (e.g., with respect to what correctness and completeness are defined), and the central notion… that of membership of a document in a category is, due to its subjective character, inherently nonformalizable."

For document collections used in this work it was not possible to obtain indexing policies. And judging from the assigned concepts, there were considerably higher levels of exhaustivity and specificity than would be typical, in addition to the relatively large size of the vocabulary (see 3.3). However, because methodology for qualitative evaluation has yet to be developed, and due to limited resources, in all but one of the studies reviewed here (Golub and Lykke Nielsen 2009), the common approach to evaluation was followed, i.e. the assumption was that manually assigned classes in document collections were correct, and automatically assigned classes were compared against them.

2.4 Hierarchical subject browsing

While it has been reported that users prefer searching to browsing (Nielsen 1997; Lazonder 2003), browsing has nevertheless been claimed to have a number of advantages. It is an intuitive activity which is cognitively easier than searching, and which helps clarify an information problem (Large et al. 1999, 192). It is especially useful when users are not looking for a specific information resource, when they lack experience in performing searching, or when they are not familiar with the subject or its structure and terminology (Koch and Zettergren 1999).

Examples of Web-based services offering subject browsing include quality-controlled subject gateways such as Intute (Intute Consortium 2006a), or those provided by commercial search engines such as Yahoo! Directory (Yahoo! 2010) or Google Directory (Google 2010). However, subject browsing does not generally seem to be well supported in information services on the Web. For example, in his study on browsing strategies and implications for the design of Web search engines, Xie (1999) reports that the existing browsing features of search engines are insufficient for the needs of users. One of the possible reasons for this lack of development could be that people believe to a large extent that browsing is less useful than searching. Even within the Renardus project, an initial suspicion about potential user requirements was that end users favoured searching over browsing (Tuominen et al. 2000). After the browsing interface was built, it was shown that browsing was much preferred to searching (Koch et al. 2006). Large et al. (1999, 180) claim that users are often able to express their information needs only in very general terms and that these can be met only by incorporating both browsing and searching capabilities in information retrieval systems.

Controlled vocabularies (classification schemes, thesauri, subject heading systems) have traditionally been used in libraries and in indexing and abstracting services, in some cases since the 19th century. With the coming of the Web, new versions of vocabularies emerged within the computer science and the Semantic Web communities: ontologies and search-engine directories of Web pages. All of these vocabularies have distinct characteristics and are consequently better suited for some applications than others. For example, subject heading systems do not normally include detailed hierarchies of terms, while classification schemes consist of hierarchically structured groups of classes. In classification schemes, similar documents are also grouped together into classes, and relationships between the classes are established. Thus, they are better suited for subject browsing than other controlled vocabularies (Vizine-Goetz 1996; Koch and Zettergren 1999). This is partly confirmed by the fact that they have been used by several Web-based services, especially those providing information resources for academic users, such as the BUBL Information Service (2005), INFOMINE (2010) etc.

Different classification schemes have different characteristics. For subject browsing the following are particularly important:

1) The bigger the collection, the more depth the hierarchy should contain;
2) Hierarchically flat schemes are not effective for browsing; and,
3) Classes should contain more than just one or two documents (Schwartz 2001, 48).

Search-engine directories and other home-grown schemes on the Web, "…even those with well-developed terminological policies such as Yahoo… suffer from a lack of understanding of principles of classification design and development. The larger the collection grows, the more confusing and overwhelming a poorly designed hierarchy becomes…" (ibid., 76).

Based on this information, the following two classification schemes were chosen in this work:

1) DDC (OCLC 2010), which has been used (and updated) in libraries for more than a century now; and,
2) The Ei classification scheme (Milstead 1995), which has been used and maintained in the Compendex database (Engineering Information 2010).

3 Methodology

3.1 Evaluation

3.1.1 Browsing behaviour

In order to study the effects of browsing behaviour on classification schemes, two different types of user study were conducted. The first involved log analysis of a large Web-based service providing integrated searching and browsing access to quality-controlled web resources classified into DDC (Koch et al. 2006). Log analysis aims to interpret computer data of recorded user actions conducted on a web site over a period of time. It includes steps such as cleaning out the log files (which may be robot actions) and creating datasets and structures for analysis. This method was chosen because users do not need to be directly involved in the study: user behaviour is captured in natural conditions, and every activity inside the service is tracked. The study encompassed 16 months of usage. Purpose-built tools for log analysis were developed, since existing software packages did not support all of the required tasks.

The second study involved 40 subject experts in engineering who, given four tasks, were asked to find the most appropriate class in the Ei classification scheme and evaluate whether the top-ranked Web pages were within the topic of the task (Golub and Lykke Nielsen 2009). The study was based on web pages which had been automatically crawled and classified into the Ei scheme. The data were collected through questionnaires, logging users' browsing steps, correctness assessments, and by observation.

3.1.2 Automated classification

In order to evaluate the degree to which automated classification yield correct classes, two main methods were used. The first involved comparison of automatically assigned classes to pre-existing, manually assigned ones, employing standard evaluation measures, precision, recall and F1 (as follows) (Sebastiani 2002, 40-41) (used in Golub and Ardö 2005; Golub 2006c; Golub et al. 2006; Golub et al. 2007):

Precision = correct automatically assigned classes /
all automatically assigned classes

Recall = correct automatically assigned classes /
all manually assigned classes

F1 = 2*Precision*Recall / (Precision + Recall)

In addition to making an exact comparison between automatically and manually assigned classes, it was also possible to determine partial matches, as the Ei classification scheme has a solid hierarchical structure in which topical relatedness of classes is expressed in numbers representing the classes (class notation). The more initial digits any two classes have in common, the more related they are. For example, 933.1.2 for *Crystal Growth* is closely related to 933.1 for *Crystalline Solids*, both of which belong to 933 for *Solid State Physics*, and all three of them belong to 93 for *Engineering Physics*. Each digit represents one hierarchical level: class 933.1.2 is at the fifth hierarchical level, 933.1 at the fourth etc.

The average number of classes assigned to each document was also examined. In the context of hierarchical browsing based on a classification scheme, having too many classes assigned to a document would place one document in too many different places, which would in turn create the opposite effect of the original purpose of a classification scheme, i.e. grouping similar documents together. Several other factors were also examined, such as the number of documents being classified, whether the main concept is discovered, and so on.

The second method involved an end user study, combining evaluations of browsing behaviour and automated classification correctness together (outlined in the last paragraph of 3.1.1).

3.2 Document collections

Browsing behaviour was studied with respect to two collections of Web pages. One comprised documents that were manually classified into DDC, containing over 80,000 Web pages (Koch et al. 2006). The other consisted of about 19,000 Web pages which had been automatically crawled and classified into the Ei classification scheme (Golub and Lykke Nielsen 2009).

The easiest way to evaluate algorithms for automated classification is through a collection of documents that were previously manually classified. The collection used in Golub and Ardö (2005) contained web pages which had been manually classified into the Ei scheme. It comprised only about 1,000 documents, and bigger collections of manually classified web pages in the same subject area were not available. As a consequence, further experiments aimed at uncovering improvements to the classification algorithm were conducted on a collection of some 35,000 bibliographic records with abstracts from the Compendex database, which were also manually classified into the Ei classification scheme (Golub 2006c; Golub et al. 2007). A comparison of the performance of string-matching and machine-learning algorithms was conducted on a similar set from Compendex, comprising about 24,000 bibliographic records (Golub et al. 2006).

3.3 Engineering Information thesaurus and classification scheme (Ei)

Ei consists of two separate parts with mappings between them: a thesaurus of engineering terms, and a hierarchical classification scheme of engineering topics (Milstead 1995). In information retrieval systems, these two controlled vocabulary types have each traditionally had distinct functions: the thesaurus has been used to describe a document with as many controlled terms as possible for the purpose of allowing detailed searching, while the classification scheme has been used to group similar documents together in order to allow systematic browsing. The Ei classification scheme is hierarchical and consists of six main classes divided into 38 finer classes, which are further subdivided into 182 classes. These are subdivided even further, resulting in some 800 individual classes in a five-level hierarchy. In this work, the classification scheme was used for systematic browsing, while thesaurus terms with their mappings to the classification scheme were utilised in the classification algorithm.

In the studies by Golub (2006c), Golub et al. (2007), and Golub and Lykke Nielsen (2009), 92 classes were used. They all belonged to class 900, *Engineering, General*. The reason for choosing this group of classes was that it covers both natural sciences such as physics, and social sciences such as management. The literature of the latter tends to contain more polysemic words than the former and, as such, presents a more complex challenge for automated classification. In the study by Golub et al. (2006), six classes were selected, the ones for which there were the most documents in the document collection (see 3.2).

A major advantage of Ei for automated classification is that thesaurus descriptors are mapped to classes of the classification scheme. These mappings have been created manually and are an integral part of the thesaurus. Compared with captions (class names) alone, mapped thesaurus terms provide a rich additional vocabulary for every class: instead of having only one caption per class, there are 88 terms per class on average (for the 92 classes used in Golub 2006c; Golub et al. 2007; and, Golub and Lykke Nielsen 2009). In addition, Ei contains a large number of composite terms (3,474 in the total of 4,411 distinct terms for the 92 classes): as such, it provides a rich and precise vocabulary with the potential to reduce the risk of false hits in string-matching classification algorithms.

3.4 The string-matching classification algorithm

This section describes the string-matching classification algorithm used in Golub and Ardö (2005), Golub (2006b), Golub (2006c), Golub et al. (2006), Golub et al. (2007), and Golub

and Lykke Nielsen (2009). The algorithm classifies documents into classes of the Ei classification scheme, with the purpose of enabling browsing access to the document collection. String matching reuses the valuable work that has been invested into building a quality controlled vocabulary like Ei, containing mappings between thesaurus terms and class captions. As such, it does not require pre-classified documents for training algorithms while still maintaining a pre-defined structure suitable for subsequent systematic browsing.

Based on thesaurus terms and captions, a term list was created that served as an input to the algorithm. The list was formed as an array of triplets (for a formal description, see Ardö 2007):

Weight: **Term** (*Single-word, Boolean or Phrase*) = **Class**

The list contained class captions and thesaurus terms (**Term**), classes which they represent or map to (**Class**), and weight indicating how appropriate they are for that class (**Weight**). *Single-word* terms consisted of one word. *Boolean terms* consisted of two or more words which had to be present but could be in any order or at any distance from each other. *Phrases* also consisted of two or more words but they had to be present in the same order and at the same distance from each other.

The algorithm searches for **Terms** from a given term list in the document to be classified. If the **Term** is found, the **Class(es)** mapped to it in the term list are assigned to the document. One class can be designated by many terms, and each time a term is found, the corresponding **Weight** is added to the score for the class for that document. The scores for each class are summed up and classes with scores above a certain cut-off value (heuristically defined) are selected as the final ones for the document.

4 Results

4.1 Usage of Web-based browsing

With the purpose of determining whether hierarchical Web-based browsing is being used and, if so, how (research question **1**), a study of a large Web-based service was conducted (Koch et al. 2006). The service chosen was Renardus, which offered integrated searching and browsing access to about 80,000 quality Web pages from major European subject gateways. Both browsing and searching options were elaborately developed. The main navigation feature was browsing based on a well-established classification scheme, DDC (OCLC 2010). Browsing-support features were also provided:

1) The graphical fish-eye presentation of the classification hierarchy (for an example see http://www.it.lth.se/knowlib/renardus-log/Graph100.jpg);
2) Search entry to browsing pages, retrieving a list of all captions containing the searched-for string (“Find a different start page for browsing” in the example at http://www.it.lth.se/knowlib/renardus-log/Browse.jpg);
3) Merging web page descriptions from contributing collections;
4) Simple searching; and,
5) Advanced searching, allowing several combinations of search terms and search fields, and options to limit searches in different ways (http://www.it.lth.se/knowlib/renardus-log/Advan.jpg).

In contrast to earlier research which reported that searching is used more than browsing (Nielsen 1997; Lazonder 2003) and anecdotal indications that that might be the case, this study clearly indicated that browsing as an information-seeking activity is widely employed, given proper conditions. About 80% of all activities in Renardus were browsing activities,

whereas only 5% involved searching. One factor contributing to the dominance of browsing was that the majority of users (71%) had been referred from search engines directly to browsing pages in Renardus. The browsing pages were pages listing a specific, easily 'browseable' sub-tree of the DDC directory, with its broader, narrower and co-ordinated classes. However, users starting at the home page (22%) predominantly used the browsing part of the service as well. This, on the other hand, could be attributed to the fact that while searching option was made available, the layout of the home page invited browsing by offering the top level DDC browsing tree above the search box.

The DDC directory-style browsing was the single most dominant activity in Renardus (60%). Two-thirds of it was done in unbroken sequences, some of them surprisingly long: while the majority limited themselves to around 10 such steps, long unbroken sequences of up to 86 steps were found. These are unexpected results, as it is often assumed that people looking for information on the Web use as few clicks as necessary, switching frequently to other services and activities, and having short attention spans.

The browsing support features were also heavily used: they made up 13% of all activities. The two most frequently used support features were the graphical fish-eye display and the search entry to browsing pages, which had been designed to relieve users from the necessity of having to jump around in the hierarchy. Jumping one step up and another step down in the directory-style display was probably faster and easier than using the support features; moving farther away might possibly have been easier using the support features.

Transitions between different types of activities were rare, despite the provision of a full navigation bar on each page of the Renardus service. When they took place, it was mostly between different browsing activities (DDC directory-style browsing and browsing support features). Switching between browsing and searching occurred in 7% of the sessions, far less than was hoped.

Users starting at the homepage showed different behaviour than users coming to the service at one of its browsing pages. Those who started at the homepage performed almost twice as many activities per session, used searching pages five times as often, and visited other pages three times as often. They were a minority, but they used the service elaborately, in the way that system designers had imagined and intended. These were probably the users who went deliberately to Renardus, whereas a large part of users who started elsewhere, most often in the browsing pages, ended up there incidentally via a search engine.

That browsing is well accepted was also indicated in the second user study (Golub and Lykke Nielsen 2009**)** (see the following section). Most suggestions to improve browsing that arose from that study had already been implemented in Renardus, in the form of browsing-support features.

In conclusion, both studies lead to the hypothesis that browsing and its support features are perceived to be popular and useful in services like Renardus.

4.2 Suitability of DDC and Ei for hierarchical Web-based browsing

In order to determine whether established classification schemes are suitable for hierarchical Web-based browsing (research question **2**), two different classification schemes were examined: a general-subject scheme, DDC (Koch et al. 2006), and a subject-specific scheme, in engineering, Ei (Golub and Lykke Nielsen 2009).

Log analysis of DDC usage (Koch et al. 2006) provided several insights into the suitability of its structure and vocabulary. As reported above, DDC browsing was the most dominant activity in Renardus, which is one indication of its suitability. Analysis of a sample of 100 search queries from the log, which were submitted to a search engine, showed that most search queries matched terms in DDC captions.

Analysis of browsing jumps between different parts of the DDC directory indicated that they occurred in less than half of the sessions that showed unbroken DDC-directory browsing. In those sessions, less than two jumps were carried out on average, which is not especially high. Also, the overall mean probability of moving between DDC main classes in a session is small (3%). These results imply that DDC is suitable for browsing. The findings from the log analysis can only help create hypotheses, however, and need to be complemented by investigative sessions with users.

The user study on Ei indicated that the Ei classification scheme is also generally well suited for browsing (Golub and Lykke Nielsen 2009). The majority of participants found the right class, reported that it was quite easy finding it, and were quite certain they had done so accurately. Even so, participants' comments indicated some inadequacies in the classification scheme. The need for several improvements can be deduced from these findings:

1) Follow consistent division principles;
2) Modify captions so that they better reflect concepts that they represent; and,
3) Allow for a larger entry vocabulary, which would be of direct help in finding the ideal class fast.

4.3 Improving the string-matching algorithm

Different approaches to automated classification (research question **3**) were discussed in section 2.2; in order to investigate problems and possible improvements to the string-matching algorithm (research questions **4** and **5**), four studies were conducted (Golub and Ardö 2005; Golub 2006b; Golub 2006c; and, Golub et al. 2007).

4.3.1 Challenges and recommendations

In order to identify the challenges involved in applying a string-matching classification algorithm to a collection of web pages (research question **4**), an analysis of 70 misclassified pages was conducted (Golub 2006b). Four major types of problems were identified:

1) Class not found at all;
2) Class found but below a pre-defined cut-off value;
3) Wrong automatically assigned class; and,
4) Correct automatically assigned class which had not been manually assigned.

The following reasons for these problems were recognized, and methods for dealing with them are proposed in each case:

1) Classes were not found when the term list lacked the right terms to designate the classes. In certain cases, this was due merely to a simple form variation, or to a different ordering of a term's constituent words when the term had been treated as a phrase.

In other automated classification experiments, form variation has been addressed by stemming (reducing words to their stem or root form), often at the expense of decreased precision. One solution could be to manually introduce regular expressions to the term list. An automated solution could be to apply computational linguistics methods to create new variations of the existing terms. Cases in which the appropriate terms are missing could be dealt with by enriching the term list with synonyms for concepts, both for the ones already covered by the thesaurus, and by introducing new ones.

In a later study (Golub et al. 2007), it was shown that recall is improved by introducing new variations of terms, as well as synonyms for existing concepts, by using automated multi-word morphosyntactic analysis and synonym acquisition. In (Golub 2006c), the term list was enriched with more term types from the Ei thesaurus than in the original term list (Golub 2006b); recall was also improved.

2) Certain classes found by the algorithm were not assigned as final classes because their scores did not reach the cut-off value (see last paragraph in 3.4). Different weighting schemes and cut-offs were examined in later studies (Golub and Ardö 2005; Golub et al. 2007) and improvements have been achieved in terms of precision and F1 (see 4.3.3 for details).

3) Automatically assigned classes seemed to have been wrongly assigned as a result of three different problems. Firstly, terms found on web pages were homonyms or very distant synonyms for concepts designated by the same terms on the term list. Secondly, terms from the term list found on web pages represented *an instance of* the concept designated by the term and were not *about* such an instance (e.g. a web page that is an information service on the topic of artificial intelligence gets wrongly classified as being on the topic of information services). Thirdly, mappings between a thesaurus term and a class were too distant.

The following solutions are proposed to tackle these issues:

a. Adding context to single and ambiguous terms, e.g. by enriching them with corresponding broader terms;
b. Introducing synonyms;
c. Creating a stop-word list of homonyms that always yield incorrect classes, in order to filter them out from the start; and
d. Classifying hyperlinked web pages and comparing their classes to derive the one with the greatest frequency.

In Golub et al. (2007), significant improvements were achieved by excluding those terms from the term list that had previously been shown to find the wrong classes in most cases, as a result of the three problems described above.

4) For various minor reasons, certain classes that were assigned automatically should or could have been manually assigned. As a result of such omissions, automatically assigned classes in research studies should be also evaluated for accuracy by subject experts. Such a user study was conducted and reported in Golub and Lykke Nielsen (2009) (see 4.5).

4.3.2 HTML structural elements and metadata

The aim of the study by Golub and Ardö (2005) was to determine the importance of distinguishing between different elements of a web page in automated classification. The hypothesis was that the best results are achieved when different weights are assigned to classes, based on where the terms designating the classes are found on a web page. Four elements of web pages were studied: title, headings, internal metadata, and body text. The document collection consisted of some 1,000 Web pages in engineering, to which Ei classes had been manually assigned.

First, potential weights were obtained, using several different methods: precision and recall based on both total and partial overlap, semantic distance and multiple regression. Next, the derived weights were tested against the baseline, where all the four elements had equal weight.

It was shown that the best results were obtained when all the four elements of the web page were taken into account. However, the exact manner in which the weights for terms found in those elements were combined turned out not to be especially important: the best combination of weights was 3% better than the baseline. In the best combination of weights, great significance was given to classes that were assigned based on the title: for example, the score for one class was the sum of the score for that class found by title multiplied by 86, the score from metadata multiplied by 6, the score from headings multiplied by 5, and the score from body text.

These findings need to be examined further. One might guess that this was because web pages in the document collection were rather heterogeneous; on the other hand, they were selected by librarians for end users of an operational service and, as such, they might indicate what similar collections of web pages tend to be like. Apart from heterogeneity, the problem could be that metadata were abused (e.g., to increase ranking in search engines), or that certain tags were misused (e.g., a headings tag was used instead of using appropriate tags for making text bold, which has the same visual effect).

4.3.3 Improvements achieved on paper abstracts

The study by Golub (2006c) explored to what degree different types of terms in the Ei thesaurus and classification scheme influence the performance of automated classification. Preferred terms, their synonyms, broader terms, narrower terms, related terms, and captions were examined in combination with a stemmer and a stop-word list. The document collection comprised some 35,000 abstracts of scientific papers from the Compendex database. A subset of the Ei thesaurus and classification scheme was used, containing 92 classes from the area of General Engineering.

The results showed that preferred terms perform best, and captions perform worst. Stemming in most cases improved performance, while the stop-word list did not have a significant impact. The majority of classes were found when using all types of terms, and when using stemming: recall was 73%. The remaining 27% of classes were not found because terms designating the classes on the term list did not exist in the documents being classified. The number of terms designating a class did not in itself seem to be related to the classification performance for that class.

The study implied that all types of terms should be included on a term list in order to achieve the best recall. Higher weights could be given to preferred terms, captions and synonyms, as they yield the highest precision.

In this study, neither weights nor cut-offs were tested; instead, all the classes that were found for a document were assigned to it. In the context of hierarchical browsing based on a classification scheme, having too many classes assigned to a document would place one document in many different places, which would create the opposite effect of the original purpose of a classification scheme, i.e. grouping similar documents together. The aim of the study by Golub et al. (2007) was to further improve the classification algorithm, especially the following aims:

1) Achieve precision levels similar to the levels of recall in the previous study (Golub 2006c)**,** by applying different weights and cut-offs; and
2) Increase levels of recall achieved in the previous study (Golub 2006c)**,** by natural language processing methods.

In order to vary different parameters systematically, 14 weighting schemes evolved. They combined weights for different term types (single-word, Boolean, phrase), class types (main or optional), Ei term types (preferred, related etc.), the number of words contained in a term, and the number of times each of the words occur in other terms:

a. Baseline (all term types given equal weight);
b. Term types, with weights derived from a separate experiment (single-word terms 1, phrases 3, Boolean terms 4);
c. Terms mapping to the main class vs. optional class, with weights derived from a separate experiment (optional class 1, main class 2);
d. Weights of the previous two lists combined (weights for term type 1, 3, and 4 for a single, phrase or Boolean term multiplied by the weight for the type of class to which the term mapped – 1 or 2 for optional or main class);

e. Weights as used in an early experiment (Koch and Ardö 2000) where weights were intuitively derived, also taking into consideration the term type and mapping to the main or optional class (single and optional 1, single and main 2, Boolean and optional 2, Boolean and main 3, phrase and optional 4, phrase and main 8);
f. Ei term types, with weights derived from a separate experiment (broader 1, related 1, narrower 2, preferred 2, synonyms 3, captions 4);
g. Weights of term types and Ei term types combined (weights for term type 1, 3, and 4 for a single, phrase or Boolean term multiplied by the weight for the type of Ei term as given in the previous list);
h. Weights of term types combined with Ei term types and class mappings (weights for term type 1, 3, and 4 for a single, phrase or Boolean term multiplied by the weight for the type of class to which the term mapped – 1 or 2 for an optional or main class, and by the weight for the type of Ei term as given in list f.);
i. Modified *tf-idf*, weights calculated based on the number of words the term consisted of, and on the number of times each of its words occurred in other terms (cf. *tf-idf*, term frequency – inverse document frequency, Salton and McGill 1983, 63, 205);
j. As preceding, with phrases modified as Boolean terms – in order to study the influence of phrases and Boolean terms on precision and recall;
k. As under i., with Boolean terms modified as phrases – in order to study the influence of phrases and Boolean terms on precision and recall;
l. As modified *tf-idf*, with those weights multiplied by the weight for the type of class to which the term maps – 1 or 2 for an optional or main class;
m. As modified *tf-idf*, with those weights multiplied by the weight for the Ei term type (f.); and,
n. As above, with those weights multiplied by the weight for the type of class to which the term maps – 1 or 2 for an optional or main class.

A stop-word list and stemming were also tested. The effect of different cut-off parameters was also investigated, as follows:

a. The score for classes to be selected as final classes had to reach a minimum percentage of the sum of all the classes' scores;
b. If no class attained the required minimum score, the one with the highest score was assigned; and,
c. Score propagation, where scores for classes at deeper hierarchical levels were increased by the scores for classes at broader hierarchical levels.

It was shown that the score propagation does not yield significantly different results, while the second rule listed above, whereby at least the class with highest score is assigned, results in more documents with correctly assigned classes.

In order to further improve recall, the basic term list was enriched with new terms. These terms were extracted using multi-word morphosyntactic analysis and synonym acquisition, based on the original preferred and synonymous terms, where those two term types resulted in the best precision (Golub 2006c). Extracted synonyms were verified by a subject expert.

In conclusion, the study by Golub et al. (2007) showed that the string-matching algorithm could be enhanced in a number of ways:

1) Weights: adding different weights based on whether a term is a single-word, Boolean, or phrase, which type of class it maps to, and Ei term type (the weighting scheme listed

under f.). This improves the precision and relevance order of the assigned classes, the latter being important for browsing;
2) Cut-offs: selecting as final classes those above a certain cut-off level improves precision and F1;
3) Enhancing the term list with new terms based on morphosyntactic analysis and synonym acquisition improves recall; and,
4) Excluding terms that in most cases gave the wrong classes yields the best performance in terms of F1, where the improvement is due to increased precision levels.

4.4 The string-matching algorithm on an abstracts collection

Finally, using the above enhancements (weights, cut-offs, stemming, and stop-word removal) the performance of the string-matching algorithm on a collection of abstracts was assessed (research question **6**). At the third and second hierarchical levels the mean F1 reached up to 66% (Table 1, where 90, 91, 92 etc. represent classes at the second hierarchical level, and 901, 902, 903 etc. represent classes at the third hierarchical level).

Classes	*General*			*Management*				*Maths*		*Physics*			*Instruments*			
2nd level	*90*			*91*				*92*		*93*			*94*			
F1 (%)	65			50				**66**		51			49			
3rd level	*901*	*902*	*903*	*911*	*912*	*913*	*914*	*921*	*922*	*931*	*932*	*933*	*941*	*942*	*943*	*944*
F1 (%)	35	27	**53**	32	**36**	26	29	**59**	33	44	33	**48**	28	36	20	**44**

Table 1. Algorithm performance on a collection of abstracts at the second and third hierarchical levels

For all hierarchical levels, the best mean F1 was 38%, when only those terms that found classes correctly in the majority of cases were included on the term list. The best recall was 76%, when the basic term list was enriched with new terms (applying morphosyntactic analysis and synonyms acquisition), and precision was 99% when just those terms that had produced only correct classes were included. When using the original term list without any terms excluded, the precision of individual classes was up to 98%. For further details see Golub et al. (2007).

These results are comparable to machine-learning algorithms (see Sebastiani 2002), which are considered to perform the best, but which require training documents and are collection-dependent. Another benefit of classifying documents into classes of well-developed classification schemes is that they are suitable for subject browsing, unlike automatically-developed controlled vocabularies or home-grown directories, which are often used in document clustering and machine learning (see section 2.2).

It was also shown that different versions of the algorithm could be implemented to best suit the application in which the automatically classified document collection was used. If high recall is required, for example in focused crawling, cut-offs need not be used. If providing a directory-style browsing interface to a collection of automatically classified web pages, the pages could be ranked by relevance based on their scores. In such a directory, one would want to limit the number of web pages per class, e.g., assign only the class with the highest probable accuracy, as is done in Thunderstone's Web Site Catalog (Thunderstone 2010). Considering that, for 14 classes at the top three hierarchical levels, the mean F1 is almost twice as high as for the complete matching, this classification approach would better suit information systems in which fewer hierarchical levels are needed, like the Intute subject gateway for engineering (Intute Consortium 2006b).

4.4.1 Comparison to a machine-learning algorithm

In an exploratory study (Golub et al. 2006) the string-matching algorithm was compared to a machine-learning algorithm, support vector machine (SVM). The document collection

consisted of a subset of about 24,000 Compendex paper abstracts, classified into 6 different classes, 2 of them from class 900.

SVM on average outperformed the string-matching algorithm. The first hypothesis, that SVM would yield better recall, whereas string-matching would yield better precision, was confirmed only for one of the classes. The second hypothesis was that classification performance could be improved by confederating the two algorithms. Terms (features) used by one algorithm were combined with the other algorithm's terms in five different ways. The results showed that SVM performed best in its original setting, while recall and F1 improved for string-matching when using the SVM terms.

Since this study had already been conducted before further improvements were introduced to the string-matching algorithm (Golub et al. 2007), performance based on those improvements could not be reported until now. It was shown that the performance for two classes from class 900, when using the full term list with 8,099 terms, was better than when only preferred terms, synonyms and captions were used (Golub 2006c). When using a shortened term list containing 1,308 terms that always yielded correct classes for a similar document collection, with a 5% cut-off, the precision for both classes was 100%. These results are better than SVM in any setting. In the same setting, however, recall is less than 10%. When using the same shortened list with 1,308 terms, applying stemming and no cut-offs, the best recall that was achieved for class 903.3 is 61%. This recall is the same as in the first experiment (Golub et al. 2006) but, in the second (Golub et al. 2007), precision is higher, so F1 is consequently higher too (46%). F1 Values are still lower than when the term list is enriched with *tf-idf* centroid terms produced as part of the SVM algorithm.

While SVM, as used in this study, outperforms the string-matching approach for recall and F1, it must be borne in mind that when it comes to real-life information systems such as digital libraries, pre-classified document collections (especially of web pages) are rarely available. String-matching algorithms could be the best feasible solution in such cases.

4.5 String-matching algorithm on a Web-page collection

The study by Golub and Lykke Neilsen (2009) was carried out in order to determine the performance of the string-matching algorithm on a collection of harvested web pages, as evaluated by end users (research question **7**). It involved 40 engineering subject experts and 4 tasks, where 19,000 web pages were automatically crawled and classified into the Ei classification scheme. In each task, the participants were directed to find information on a given topic by browsing the Ei classification scheme. Once they reached the most appropriate class, they were asked to evaluate the top ranked web pages based on their relevance to the topic of the task.

As seen from Table 2, the ten top-ranked web pages in each of the four classes were, on average, deemed partly correct (1 correct choice, 2 partly correct and 3 incorrect). A major problem with determining whether a web page is in the right class or not is that there were large differences among participants in their judgements: a number of web pages were evaluated as correct, partly correct and incorrect by different participants. It seems likely that a considerable part of the problem is the issue of "aboutness" and related subjectivity in deciding which topic a document is dealing with.

Task	Corresponding class	Evaluation
Particle accelerators	*932.1.1*	1.8
Magnetic instruments	*942.3*	1.8
Differentiation and integration	*921.2*	2.0
Professional organizations in the field of engineering	*901.1.1*	2.5
Average		2.0

Table 2. Correctness of automatically assigned standard reference classes for each search topic

As in the case of browsing, evaluations differed between the four tasks (Table 2). This agrees with the previous results of the algorithm's performance, based on a pre-classified collection of paper abstracts, where it was shown that certain classes show better performance than others (Golub et al. 2007). The worst results in both studies were gained for class 901.1.1 (Societies and institutions), which can be attributed to the fact that only one term exists for this class on the term list. In addition, most terms designating the other three classes are rather field-specific and less ambiguous than the one term designating class 901.1.1 (*societies @and institutions*).

5 Concluding remarks

In this work, it is shown that hierarchical Web-based browsing is widely used and that well-developed classification schemes such as DDC and Ei are suitable for the task. In the context of browsing, three main approaches to automated classification are recognized. In order to provide good browsing structures based on the results of the complex and much-researched automated classification algorithms discussed in this paper, machine learning and document clustering approaches would need to employ suitable controlled vocabularies. Improvements were made on the string-matching approach in several ways, and evaluation of these improved techniques showed that the results are comparable to those of state-of-the-art machine-learning algorithms, especially for certain classes and applications.

While subject browsing was shown to be useful in a large Web-based service, further investigation is required to determine to what degree it is suitable for various tasks. Some controlled vocabularies are being modified for new purposes in the online environment, to which adjustments have been proposed in the literature and indicated in this work; however, more research is required on what controlled vocabularies need to be like in order to support browsing. Moreover, while Ei proved to be reasonably suitable for automated classification, further study is required on which characteristics of controlled vocabularies are in general beneficial for automated classification.

Given that there are recognized difficulties in evaluation, it is difficult to estimate to what degree modern automated classification tools are applicable in operative information systems. The subjectivity in the correct interpretation of a document's subject matter, as has been widely discussed in the literature, has been demonstrated by the findings of this work. Evaluation results depend on a number of factors such as document collection, application context, and user tasks. The methodology for the evaluation of automated classification, including a comprehensive review of the different factors involved, should be a major area for future research. This could perhaps be accomplished most effectively through a triangulation of standard collection-based evaluation and user studies.

Acknowledgements

My deepest gratitude goes to my supervisors Anders Ardö and Traugott Koch for their resolute support and mentoring throughout the four years of my PhD, resulting in the research presented here. Many thanks to all of the other researchers who co-authored the papers mentioned. Special thanks to the anonymous reviewers whose help has given this compilation of research a much better structure and, hopefully, has made it more easily readable.

References

Ardö, Anders. 2007. Crawler internal operation. Available at: http://combine.it.lth.se/documentation/DocMain/node6.html

BUBL Information Service. 2005. Centre for Digital Library Research, Strathclyde University, Glasgow. Available at: http://bubl.ac.uk/

Chan, Lois Mai. 1994. *Cataloging and classification: an introduction*. 2nd ed. McGraw-Hill, New York.

Chen, Hao, and Dumais, Susan. 2000. Bringing order to the Web: automatically categorizing search results. In *Proceedings of the ACM International Conference on Human Factors in Computing Systems, Den Haag*, pp. 145-152.

Engineering Information. 2010. *Compendex.* Engineering Information, Elsevier. Available at: http://www.ei.org/compendex

Golub, Koraljka. 2006a. Automated subject classification of textual web documents. *Journal of Documentation*, Vol. 62 No. 3, pp. 350-371.

Golub, Koraljka. 2006b. Automated subject classification of textual web pages, based on a controlled vocabulary: challenges and recommendations. *New Review of Hypermedia and Multimedia*, Vol. 12 No. 1, pp. 11-27.

Golub, Koraljka. 2006c. The role of different thesaurus terms in automated subject classification of text. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, Hong Kong, China, 18-22 December, pp. 961-965.

Golub, Koraljka, and Ardö, Anders. 2005. Importance of HTML structural elements and metadata in automated subject classification. In *Proceedings of the 9th European Conference on Research and Advanced Technology for Digital Libraries*, Vienna, Austria, 18-23 September, pp. 368-378.

Golub, Koraljka, Ardö, Anders, Mladenić, Dunja, and Grobelnik, Marko. 2006. Comparing and combining two approaches to automated subject classification of text. In *Proceedings of 10th European Conference on Research and Advanced Technology for Digital Libraries*, Alicante, Spain, 17-22 September, pp. 467-470.

Golub, Koraljka, Hamon, Thierry, and Ardö, Anders. 2007. Automated classification of textual documents based on a controlled vocabulary in engineering. *Knowledge Organization*, Vol. 34 No.4, pp. 247-263.

Golub, Koraljka, and Lykke Nielsen, Marianne. 2009. Automated classification of web pages in hierarchical browsing. *Journal of Documentation*, Vol. 65 No. 6, pp. 901-925.

Google. 2010. *Google Directory*, Google. Available at: http://directory.google.com/

INFOMINE: scholarly Internet resource collections. 2010. Library of the University of California. Available at: http://infomine.ucr.edu/

International Organization for Standardization. 1985. *Documentation – Methods for examining documents, determining their subjects, and selecting index terms*: *ISO 5963,* Geneva, International Organization for Standardization.

Intute Consortium. 2006a. *Intute.* Available at: http://www.intute.ac.uk/

Intute Consortium. 2006b. *Intute: Science, engineering and technology – engineering*. Available at: http://www.intute.ac.uk/sciences/engineering/

Jain, Anil K., Murty, Narasimha M., and Flynn, Patrick J. 1999. Data clustering: a review. *ACM Computing Surveys,* Vol. 31 No. 3, pp. 264-323.

Koch, Traugott and Ardö, Anders. 2000. Automatic classification of full-text HTML-documents from one specific subject area. (EU Project DESIRE II D3.6a, Working Paper 2). Available at: http://www.mpdl.mpg.de/staff/tkoch/publ/DESIRE36a-WP2.html

Koch, Traugott, Golub, Koraljka and Ardö, Anders. 2006. Users browsing behaviour in a DDC-based web service: a log analysis. *Cataloging & Classification Quarterly*, Vol. 42 No. 3/4, pp. 163-186.

Koch, Traugott and Zettergren, Ann-Sofie. 1999. Provide browsing in subject gateways using classification schemes. (EU Project DESIRE II.) Available at http://www.mpdl.mpg.de/staff/tkoch/publ/class.html

Lancaster, Frederick Wilfrid. 2003. *Indexing and abstracting in theory and practice.* 3rd ed. Facet, London.

Large, Andrew, Tedd, Lucy, and Hartley, Richard. 1999. *Information seeking in the online age.* K. G. Saur, London etc.

Lazonder, Ard W. 2003. Principles for designing Web searching instruction. *Education and Information Technologies* 8 (June 2003), pp. 179-193.

Lewis, David D., Yang, Yiming, Rose, Tony G., and Li, Fan. 2004. RCV1: a new benchmark collection for text categorization research. *The Journal of Machine Learning Research,* 5, pp. 361-397.

Markey, Karen. 1984. Interindexer consistency tests: a literature review and report of a test of consistency in indexing visual materials. *Library & Information Science Research,* 6, pp. 155-77.

Milstead, Jessica, ed. 1995. *Ei thesaurus.* 2nd ed. Engineering Information Inc., Hoboken, NJ.

Moens, Marie-Francine. 2000. *Automatic indexing and abstracting of document texts.* Kluwer, Boston.

Nielsen, Jakob. 1997. Search and you may find. *Jakob Nielsen's Alertbox*, July 15. Available at: http://www.useit.com/alertbox/9707b.html

OCLC. 2010. *Dewey Services.* Available at: http://www.oclc.org/dewey/

Olson, Hope A., and Boll, John J. 2001. *Subject analysis in online catalogs.* 2nd ed., Libraries Unlimited, Englewood, CO.

Salton, Gerard and McGill, Michael J. 1983. *Introduction to modern information retrieval*. McGraw-Hill, Auckland.

Schwartz, Candy. 2001. *Sorting out the Web: approaches to subject access.* Ablex, Westport, CT.

Sebastiani, Fabrizio. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, Vol. 34 No. 1, pp. 1-47.

Svenonius, Elaine. 1997. Definitional approaches in the design of classification and thesauri and their implications for retrieval and for automatic classification. In *Proceedings of the Sixth International Study Conference on Classification Research*, pp. 12-16.

Svenonius, Elaine. 2000. *The intellectual foundations of information organization.* MIT Press, Cambridge, MA.

Tennis, Joseph. 2009. Three creative tensions in document interpretation theory set as evidence of the need for a descriptive informatics. *Knowledge Organization*, Vol. 36 No. 4, pp. 190-199.

Thunderstone. 2010. About the Thunderstone Web site catalog. Available at: http://search.thunderstone.com/texis/websearch/about.html

Toth, Erzsébet. 2002. Innovative solutions in automatic classification: a brief summary. *Libri*, Vol. 25, No. 1, pp. 48-53.

Tuominen, Kimmo, Kanner, Janne, Miettinen, Manne, and Heery, Rachael. 2000. User requirements for the broker system: Renardus project deliverable D1.2.

Vizine-Goetz, Diane. 1996. Using library classification schemes for Internet resources. *OCLC Internet Cataloging Project Colloquium.* Available at: http://Webdoc.sub.gwdg.de/ebook/aw/oclc/man/colloq/v-g.htm

Xie, Iris H. 1999. Web browsing: current and desired capabilities. *20th Annual National Online Meeting, 18-20 May, New York, US*, pp. 523-37.

Yahoo! 2010. *Yahoo! Directory.* Available at: http://dir.yahoo.com/

Yang, Yiming. 1999. An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, Vol. 1 No. 1/2, pp. 67-88.