Computational Linguistics and Chinese Language Processing
Vol. 12, No. 2, June 2007, pp. 155-174

Hierarchical Web Catalog Integration with Conceptual Relationships in a Thesaurus

Ing-Xiang Chen*, Jui-Chi Ho*, and Cheng-Zen Yang*

Abstract

Web catalog integration has become an integral aspect of current digital content management for Internet and e-commerce environments. The Web catalog integration problem concerns integration of documents in a source catalog into a destination catalog. Many investigations have focused on flattened (one-dimensional) catalogs, but few works address hierarchical Web catalog integration. This study presents a hierarchical catalog integration (EHCI) approach based on the conceptual thesauri extracted from the source catalog and the destination catalog to improve performance. Experiments involving real-world catalog integration are performed to measure the performance of the improved hierarchical catalog integration scheme. Experimental results demonstrate that the EHCI approach consistently improves the average accuracy performance of each hierarchical category.

Keywords: Hierarchical catalog integration, conceptual relationships, thesaurus, Support Vector Machines (SVMs)

1. Introduction

Automatically integrating various information sources is pertinent for many real applications given the large, and still rapidly growing, amount of information available. For instance, an on-line service provider may merge various catalogs from other on-line vendors into its local catalog to provide customers with versatile content, and a Web portal may also have to integrate different Web catalogs from other portals to provide increasingly abundant information services to users [Agrawal and Srikant 2001]. In these examples, users can gain more relevant and organized information in an integrated catalog. They can also save considerable time, because they do not need to browse different Web catalogs. According to

* Dept. of Computer Sci. and Eng., Yuan Ze University, 135 Yuan-Tung Rd., Chungli, 320, Taiwan.
Tel.: +886-3-4638800 ext: 2361 Fax: +886-3-4638850.
The author for correspondence is Cheng-Zen Yang.
E-mail: czyang@syslab.cse.yzu.edu.tw

[Received March 18, 2007; Revised June 23, 2007; Accepted July 2, 2007]

previous studies [Keller 1997; Stonebraker and Hellerstein 2001; Kim *et al.* 2002; Marrón *et al.* 2003], Web catalog integration has attracted much research interest.

Web catalog integration is not just a straightforward classification task [Agrawal and Srikant 2001]. Exploring implicit source information can effectively improve the integration accuracy [Agrawal and Srikant 2001]. Many methods for enhancing catalog integration performance have been proposed so far. The most important approach, called ENB, enhances the Naive Bayes classifiers with implicit source information. Other state-of-the-art approaches, including Support Vector Machines (SVMs) [Sarawagi *et al.* 2003; Tsay *et al.* 2003; Zhang and Lee 2004a; Chen *et al.* 2005; Chen *et al.* 2006; Ho *et al.* 2006] and the Maximum Entropy model [Wu *et al.* 2005], have been also presented to elevate the performance of Web catalog integration, and they further outperform the ENB approach.

Past studies in text classification [MacCallum *et al.* 1998; Dumais and Chen 2000] have indicated that exploiting a hierarchical structure can bring strong advantages over using a flattened structure in classification. [MacCallum *et al.* 1998] presented a probabilistic framework, and a *shrinkage* approach was proposed to improve text classification in a hierarchy of classes. Experimental results indicate that hierarchical text classification with large numbers of features (feature set > 10000) can obtain better average accuracy performance than flattened text classification. However, the shrinkage approach may either have no effect or hurt slightly in some classes with a large amount of training data [MacCallum *et al.* 1998].

Previous hierarchical data integration studies [Doan *et al.* 2002; Rajan *et al.* 2005] examined the hierarchical structures of the destination catalog are studied to improve the accuracy of catalog integration. [Doan *et al.* 2002] extracted the domain constraint features obtained from the neighboring nodes to enhance the mapping of ontological data. [Rajan *et al.* 2005] developed a maximum likelihood-based framework that exploits the hierarchical structure of categories, and examined four mapping scenarios. Experimental results have demonstrated that hierarchical relationships in the destination catalog are effective in catalog integration. Some source class labels can further be integrated into the destination catalog as new classes to maintain a new hierarchy.

However, hierarchical relationships of the categories and subcategories between the source and destination catalogs have not been investigated in the previous work. Moreover, experimental results indicate that the previously proposed approaches only integrate the data into the leaf nodes of the destination catalog. Although past methods for conventional text classification and hierarchical catalog integration can benefit from using a hierarchical structure, they only address the hierarchical structure in the destination categories and do not consider the differing hierarchical structures in the source and destination catalogs. Hence, this work performs some pilot studies for the hierarchical catalog integration problem by

considering the implicit information embedded in the hierarchical structure of both the source and destination catalogs. The pilot experimental results reported in Chen *et al.* [2006] indicate that the implicit hierarchical information does indeed contribute to the hierarchical Web catalog integration problem.

While extending the results of our previous pilot study, this work presents an enhanced hierarchical catalog integration (EHCI) approach with conceptual relationships extracted from the source and destination catalog thesauri to improve the integration performance. An EHCI approach based on SVM was adopted in these experiments due to its good classification performance. To demonstrate the effectiveness of EHCI, its performance is compared with that of a simple hierarchical catalog integration approach (SHCI) based on previous hierarchical classification studies [Dumais and Chen 2000; Sun and Lim 2001; Sun *et al.* 2003; Vural and Dy 2004].

Results of experiments with real-world catalogs reveal that the EHCI approach consistently raises the accuracy of hierarchical Web catalog integration in almost all hierarchical levels in both Yahoo!-to-Google and Google-to-Yahoo! catalog integration. These results also demonstrate that EHCI attains an average accuracy improvement of 11.1% in Yahoo!-to-Google catalog integration, and 21.6% in Google-to-Yahoo! catalog integration. The results further indicate that hierarchical catalog integration can be effectively improved by enhancing the conceptual relationships discovered from the hierarchical thesauri.

The remainder of this paper is organized as follows. Section 2 reviews the related studies of catalog integration. Section 3 then describes in detail the hierarchical Web catalog integration and the enhanced hierarchical integration approach. Next, Section 4 shows the environmental settings and discusses the experimental results. Finally, conclusions are drawn in Section 5, along with recommendations for future research.

2. Related Work

Most methods proposed for solving the catalog integration problem have been based on a flattened structure, implying that the categories in a catalog are isolated and lack hierarchical relationships. Agrawal and Srikant were the first to study this problem in 2001, and presented an enhanced Naive Bayes approach (ENB) to improve the integration accuracy by exploiting implicit information from the source catalog [Agrawal and Srikant 2001]. Experimental results involving real-world catalogs indicate that ENB can achieve an average accuracy improvement of more than 14%. Their promising results reveal that exploiting implicit source information indeed benefits the accuracy for automated catalog integration.

Several algorithms have been proposed in the past few years to increase the accuracy of catalog integration based on a flattened structure. Since SVM has presented superior

performance in classification problems [Dumais *et al.* 1998; Joachims 1998; Yang and Liu 1999; Rennie and Rifkin 2001], many related studies have also adopted the SVM classifiers with different strategies to extract the implicit information and improve the integration accuracy. These SVM-based integration approaches include a cross-training technique for SVM classifiers (SVM-CT) [Sarawagi *et al.* 2003], a topic restriction strategy (SVM-TR) [Tsay *et al.* 2003], a cluster shrinkage approach (CS-TSVM) [Zhang and Lee 2004a], and an iterative approach with pseudo-relevance feedback (SVM-IA) [Chen *et al.* 2005]. Most of these approaches employing the SVM classifiers were found to have higher accuracy then ENB.

In addition to the SVM-based approaches, some state-of-the-art investigations have also been presented to enhance the catalog integration accuracy with a flattened structure. Zhang and Lee proposed a co-bootstrapping approach with boosting to obtain the optimal combination of heterogeneous weak hypotheses without adjusting feature weights manually [Zhang and Lee 2004b]. Wu *et al*. first extracted the source hierarchical information and then applied the Maximum Entropy model to increase the accuracy of catalog integration in a flattened structure. Their experimental results showed that their approach is more accurate than ENB.

Most previous catalog integration studies adopted a flattened structure to simplify the catalog integration problem, thus neglecting the hierarchical relationships among the categories. Since previous studies on text classification problems have reported that a hierarchical structure can improve performance, an approach called *shrinkage* was presented to further improve the Bayesian classifiers in hierarchical text classification [MacCallum *et al.* 1998]. With the shrinkage-based approach, the parameter estimation of a node is smoothed by interpolation from the parent nodes, thus significantly reducing the number of prediction errors in hierarchical text classification.

Experimental results indicate that the accuracy performance of the method of MacCallum *et al*. [1998] can be raised by shrinking each leaf node with linear interpolation of the parent nodes in the destination hierarchy. However, the classification is based on the same hierarchy, instead of considering both the source and the destination hierarchies, respectively. Therefore, the original algorithm may need to be modified for application to hierarchal Web catalog integration.

Rajan *et al*. [2005] presented a two-stage mapping and integration approach, and discussed four integration scenarios. They comprehensively investigated their hierarchical catalog integration scheme using a maximum likelihood approach, and found that its integration performance is very promising, particularly in one-to-many mapping (Scenario 3). Rajan *et al*. further demonstrated that the hierarchical structure of the destination catalog is helpful in improving integration accuracy in different data sets. However, the implicit

information in the source hierarchy has not been utilized in this work.

The hierarchical relationships between the source catalog and the destination catalog requires further investigation when considering hierarchical Web catalog integration. Chen *et al*. preliminarily explored the effectiveness of a hierarchical catalog integration scheme with the consideration of both the source catalog and the destination catalog [Chen *et al.* 2006]. Their experimental results indicated a consistent improvement in accuracy of real-world Web catalog integration over the EHCI approach. Although the performance improvements are significant, the integration effectiveness based on a hierarchical structure has not been comprehensively studied. The following sections first define the problem, and then describe the ECHI approach in detail.

3. Hierarchical Web Catalog Integration

The integration process of the hierarchical catalog integration problem involves two hierarchical catalogs. Figure 1 illustrates the integration process in which the source catalog S with a set of m categories $S_1, S_2,\ldots, S_m$, is integrated into the destination catalog D with a set of n categories $D_1, D_2,\ldots, D_n$. These categories may have subcategories, such as S_{11}, D_{11} and D_{121}.

The integration process in Figure 1 is performed by merging each document d_i in S into a correspondent destination category in D. Thus, for each directory in the hierarchy, the training documents trained as directory classifiers and local classifiers are utilized to help integrate each document d_i into a corresponding directory. Only the documents integrated into the corresponding level categories and subcategories are regarded as correctly integrated.

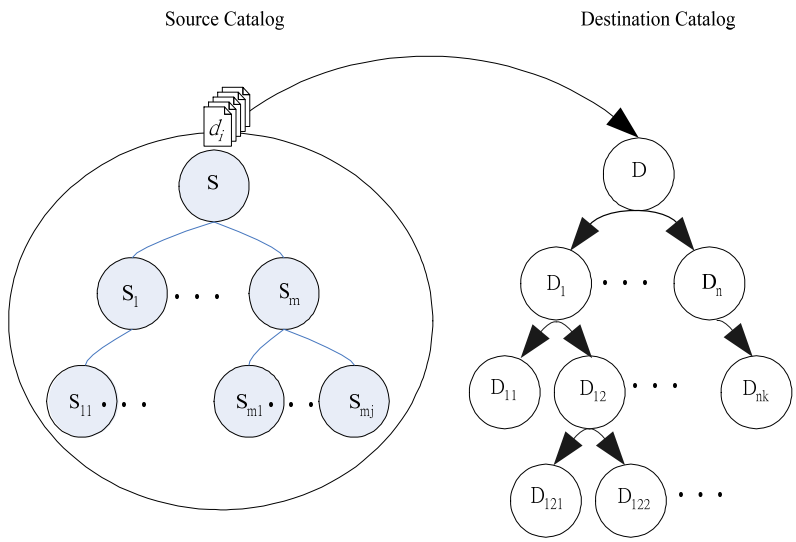

Figure 1. The process of hierarchical catalog integration.

This study adopts SVM classifiers with linear kernel functions [Yang and Liu 1999], $f: X \in R^n \rightarrow R$ to locate a hyperplane that can separate the positive examples, $f(x) \geq +1$, from the negative examples, $f(x) \leq -1$. The linear function is in the form $f(x) = (x,b) + b = \sum_{i=1}^{n} w_i x_i + b$ where $(w,b) \in R^n \rightarrow R$. The linear SVM is trained to determine the optimal values of w and b such that $||w||$ is minimized. These trained SVM classifiers are employed in the simple hierarchical catalog integration (SHCI) approach and the enhanced hierarchical catalog integration (EHCI) approach in hierarchical catalog integration. The SHCI approach and the EHCI scheme are described as follows.

3.1 The Simple Hierarchical Catalog Integration (SHCI) Approach

In SHCI, the SVM classifiers are trained with the training documents coming from the destination catalog and are used to integrate the test documents from the source catalog into the destination catalog. Whether a training document is considered a positive document or a negative document depends on its subordinate relationship to each destination category. Referring to Sun and Lim [2001] and Sun *et al.* [2003], the destination catalog was designed with two classifiers at every category node, namely a directory classifier and a local classifier.

The directory classifiers were designed to categorize the source documents into different category and subcategory trees. The directory classifiers are trained with equal numbers of positive and negative examples. The positive examples were chosen from the categories and their subcategories where the documents were located. The negative examples were selected from the remaining categories and their subcategories under the same level. The local classifiers were designed to classify the source documents further into different destination levels in each category tree. The local classifiers in each level were trained with the positive examples chosen from each destination level, and the negative examples selected from the subcategories under that level.

In real-world Web catalogs, a document may be integrated into more than one category. Therefore, a "one-against-rest" strategy was adopted to extend the binary SVM classifiers and solve the multi-class catalog integration problem. This study uses the SHCI approach as a baseline for hierarchical catalog integration, and considers the performance improvement of the SVM classifiers resulting from the enhancement of conceptual relationships in thesauri.

3.2 Conceptual Relationships in Web Thesaurus

Foskett utilized a thesaurus as a dictionary and a reference for classification [Foskett 1997]. A thesaurus can be defined as a set of related terms in a given domain knowledge, and these related terms are the basic semantic units for conveying concepts [Wikipedia: thesaurus]. Since a hierarchical thesaurus defines broad and narrow terms, its classification system can be considered a vocabulary hierarchy. Likewise, the child nodes in a hierarchical Web catalog

structure generally comprise related terms to express the classified concepts of the parent nodes, and so the classified terms in a hierarchical Web catalog can be treated as a hierarchical thesaurus. Figure 2 shows an example in which the "Automotive" category in Yahoo! Web catalog is categorized like a hierarchical thesaurus with some conceptual relationships in the hierarchy.

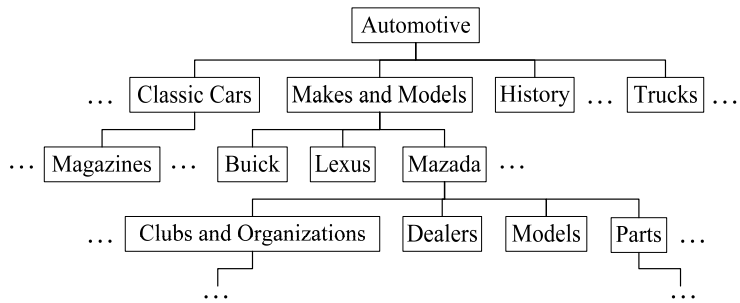

Figure 2. The illustration of a Web thesaurus in Yahoo! catalog

In Figure 2, the term "Automotive" is the thesaurus root, which expresses a broad term in the hierarchy, and has different narrow terms to define different types of "Automotive". Narrower terms are defined down to the leaf nodes in the hierarchy. In the Web catalog hierarchy, the conceptual relationships can be extracted from the hierarchical thesaurus and can construct different semantic concepts. Therefore, different domain knowledge can be extracted from the Web catalog hierarchy, thus enhancing the performance of the SVM classifiers.

3.3 Enhanced Hierarchical Catalog Integration (EHCI) Scheme

To elevate the integration performance, a weighting formula, Equation (1), is designed to exploit the conceptual relationships from the hierarchical Web thesaurus, where the terms in different category levels are extracted as label features. Equation (1) calculates the feature weight of each document, $FeatureWeight_{(x,\ d)}$, where L_i denotes the relevant label weight assigned exponentially as $1/2^i$, f_x represents the occurrence ratio of feature x in the document, and λ indicates the magnitude relation of the label weight. In Equation (1), the weight of each thesaurus is exponentially decreased and accumulated based on the increased levels, where n denotes the depth of a document in the hierarchy. If feature x appears in the label feature, then L_x is denoted as the label weight with the level where x is located. Otherwise, L_x=0. Consequently, Equation (1) is applied to both the source and destination hierarchies to represent the semantic concepts obtained from the source category labels and the destination category labels.

$$FeatureWeight_{(x,\ d)} = \lambda \times \frac{L_x}{\sum_{i=0}^{n} L_i} + (1-\lambda) \times f_x \quad (1)$$

Table 1. The label weights assigned for different hierarchical levels

Hierarchical Level	Label Weight
Document Level (L_0)	$1/2^0$
One Level Upper (L_1)	$1/2^1$
Two Levels Upper (L_2)	$1/2^2$
…	…
n Levels Upper (L_n)	$1/2^n$

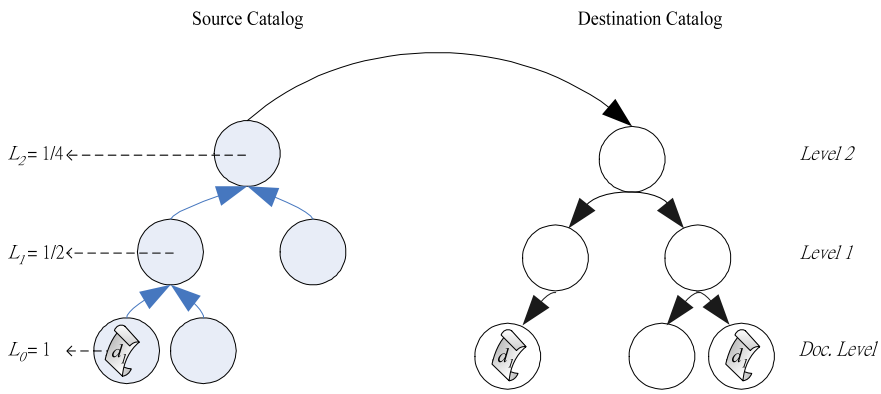

Figure 3. The process of the enhanced hierarchical catalog integration

In Equation (1), L_i further denotes the label weight at a depth of i. The label weight falls from the document level (i=0) to top level n. This thesaurus weighting method can be utilized to transform the conceptual relationships of the hierarchical source categories, and add them into the test documents. Table 1 lists the weights of different hierarchical labels, where L_0 denotes the document level; L_1 represents one level above, and so on down to L_n representing n levels above.

Similarly, the EHCI scheme is used in the destination catalog to build enhanced classifiers in destination categories. With the enhancement of the features and native category label information, the classifiers can thus be trained to be more distinctive to classify the documents into the correct categories. The weights of the features and native category label information in the destination catalog are also calculated according to Equation (1). The threshold λ is set with different values from 0 to 1 to find the optimized weights for the source thesauri to enhance the destination classifiers, as are the values of λ set in the native destination category. Moreover, the features occurring in the upper categories are removed to avoid misleading integration in the subcategories.

Figure 3 displays a three-level example to demonstrate the concept of the EHCI approach. In the source catalog, the hierarchical thesaurus information is added to the test documents with different label weights accumulated upward from their current categories to the top-level category according to the weighting formula. In the destination catalog, the test documents are integrated into the destination categories based on the EHCI integration scheme. Figure 3 also indicates that a document d_1 may be integrated into more than one destination category.

3.4 Enhanced Catalog Integration Process

Since a Web document generally comprises HTML tags, script codes and texts, the HTML tags and scripts codes are eliminated, and only the texts obtained after retrieving the Web documents from both the source and destination catalogs are kept. In the preprocessing stage, the texts are segmented into terms by removing the stopwords and stemming the terms with the Porter Stemmer [Porter 1980]. The weight of each stemmed term x is assigned by $TFx/\sum TFi$, where i denotes the number of the stemmed terms in each document. This preprocessing flow and feature weight strategy is applied to both the SHCI and EHCI schemes.

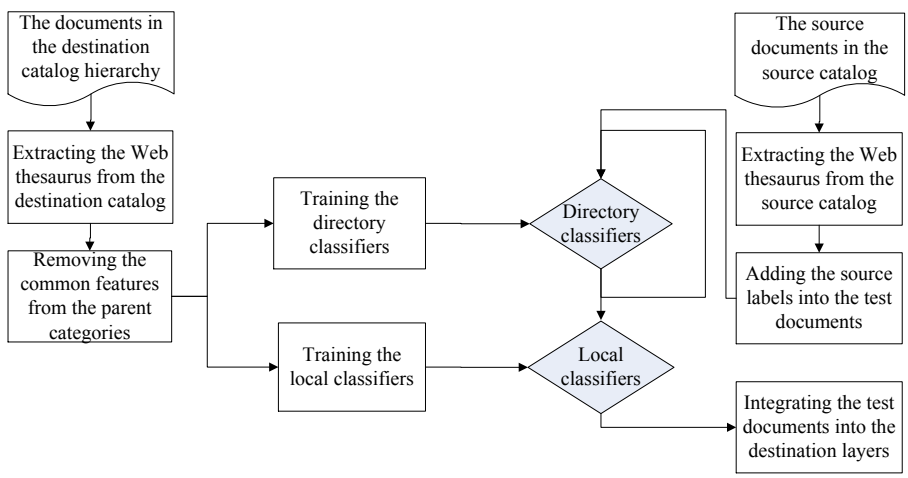

Figure 4. The process of enhanced hierarchical catalog integration

Figure 4 shows the process of hierarchical catalog integration with the EHCI scheme. In the integration process, the terms transformed from the test documents are added with the source catalog labels based on Equation (1). Similarly, the terms transformed from the documents in the destination categories are trained using the labels extracted from the destination catalog. To establish the directory classifiers and local classifiers in the destination catalog, the common features in the parent categories are removed in the training stage to avoid building ambiguous classifiers.

In Figure 4, the directory classifiers are trained with the positive documents from their categories and subcategories to represent the classifiers of the category trees. The local classifiers are trained by the positive documents of the same levels to represent the classifiers of their local levels. The selection of negative examples in the directory classifiers and the local classifiers is similar to the SHCI approach as described in Section 3.1. The test documents are then integrated into the destination categories through both the directory classifiers and the local classifiers. The integration process is finished when all the test documents from their source categories are integrated into the designated destination categories.

4. Experiments and Discussion

Experiments were performed involving real-world catalogs from both Yahoo! and Google to examine the performance of the EHCI schemes with SVM^{light} [Joachims 2002]. The average integration performance with different λ values between 0 and 1 were compared. The results with the optimal λ value are listed in detail. Experimental results indicate that the EHCI approach consistently enhances the SVM classifiers in almost all levels and boosts the integration accuracy of a hierarchical structure. The following subsections describe the data sets and the experimental results.

4.1 Data Sets

Five categories were extracted from Yahoo! and Google. Table 2 shows the statistics of our experimental data including the number of hierarchical classes, the training documents and the test documents in these five categories. The experimental data were collected after neglecting the documents that could not be retrieved and removing the documents with error messages. The stopword list in Frakes and Baeza-Yates [1992] was adopted to remove the stopwords in preprocessing. Over 38,000 terms were employed for training and testing after removing the stopwords and stemming. As in [Agrawal and Srikant 2001], documents appearing in only one catalog were used as the training documents in the destination catalog D, and the common documents were adopted as the test documents in the source catalog S.

Table 2. The experimental data collected from the Google catalog

Category	Google	|G-Y|	|G Class|	|G Test|	Yahoo!	|Y-G|	|Y Class|	|Y Test|
Autos	…/Autos/…	1094	312	437	…/Automotive/…	1823	148	404
Movies	…/Movies/…	5174	1165	1340	…/Movies_Film/…	7776	1035	1211
Outdoors	…/Outdoors/…	2308	523	224	…/Outdoors/…	1724	100	177
Photo	…/Photography/…	615	158	206	…/Photography/…	1399	80	175
Software	…/Software/…	5693	1185	683	…/Software/…	1940	109	646
Total		14884	3343	2890		14662	1472	2613

A set of 1,472 classes in the Yahoo! catalog and a set of 3,343 classes in the Google catalog were organized according to the original hierarchy to a depth of six levels as shown in Table 2. The test documents were chosen by cross-referencing the documents of Yahoo! with those of Google. Table 2 indicates that the numbers of test documents in Yahoo! and Google were different, in the sense that some test documents may appear in more than one class simultaneously. The training documents of the Yahoo! catalog and the Google catalog were accumulated by subtracting the common documents in the other catalog. In this experiment, the documents were integrated both from Yahoo! into Google and from Google to Yahoo!.

Tables 3 and 4 further describe the number of the hierarchical classes, the training documents, and the test documents of six levels in the Google and Yahoo! catalogs. Since most of the sixth levels contain less than ten documents, the hierarchies were only retrieved down to the sixth level, and any documents below the sixth level were merged upward to the sixth level. Tables 3 and 4 indicate that the numbers of some Level 1 classes were zero, meaning that the destination category contained no Level 1 test documents. This experiment only considered the documents that were correctly integrated into the destination categories, thus we list the number of classes with common test documents.

Table 3. The experimental data collected from the Google catalog

	Level 1	Level 2	Level3	Level 4	Level 5	Level 6	Total
Class # in Autos	0	14	98	148	46	6	312
Training doc.# in Autos	0	144	422	389	127	12	1094
Test doc. # in Autos	0	86	218	111	19	3	437
Class # in Movies	1	27	115	700	245	77	1165
Training doc.# in Movies	3	136	2581	1554	718	182	5174
Test doc. # in Movies	0	131	524	348	302	35	1340
Class # in Outdoors	1	23	114	111	104	170	523
Training doc.# in Outdoors	1	104	594	376	434	799	2308
Test doc. # in Outdoors	0	40	76	69	24	15	224
Class # in Photo	0	9	29	50	52	18	158
Training doc.# in Photo	0	28	172	227	141	47	615
Test doc. # in Photo	0	26	88	59	25	8	206
Class # in Software	1	59	281	352	306	186	1185
Training doc.# in Software	2	547	1784	1656	1189	515	5693
Test doc. # in Software	2	29	149	241	157	105	683

Table 4. The experimental data collected from the Yahoo! catalog

	Level 1	Level 2	Level3	Level 4	Level 5	Level 6	Total
Class # in Autos	1	24	61	39	17	6	148
Training doc.# in Autos	56	490	575	467	186	49	1823
Test doc. # in Autos	11	126	119	101	38	9	404
Class # in Movies	1	27	91	195	584	137	1035
Training doc.# in Movies	2	653	992	2210	3260	659	7776
Test doc. # in Movies	0	140	180	353	404	134	1211
Class # in Outdoors	1	26	47	17	5	4	100
Training doc.# in Outdoors	63	455	815	305	25	61	1724
Test doc. # in Outdoors	0	44	114	18	0	1	177
Class # in Photo	1	18	28	14	17	2	80
Training doc.# in Photo	28	266	453	138	496	18	1399
Test doc. # in Photo	1	72	78	19	5	0	175
Class # in Software	1	15	24	24	26	19	109
Training doc.# in Software	50	366	488	489	364	183	1940
Test doc. # in Software	3	146	133	155	174	35	646

Table 5. The analysis of common classes between Google and Yahoo!

	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6
Common class # in Autos	-	2	39	1	0	0
Common class # in Movies	-	16	24	9	6	1
Common class # in Outdoors	-	5	3	2	3	0
Common class # in Photo	-	4	2	1	3	0
Common class # in Software	-	7	6	8	8	3

Since hierarchical catalog integration is not like hierarchical text classification on the basis of the same hierarchy, the structure of the source hierarchy can be very different from the structure of the destination hierarchy. Table 5 further analyzes the number of common classes in different levels between the Yahoo! catalog and the Google catalog. Table 5 indicates that the number of common classes from Level 2 to Level 6 was very small. For example, the Level 2 category of "Autos" contains only two common classes (chats_and_forums and makes_and_models) between the Google catalog (14 classes) and the Yahoo! catalog (24 classes).

In addition to the common classes in the same Level 1 categories, the common classes in different Level 1 categories were also analyzed. The results reveal that the different Level 1 categories had very few common classes or even no common classes in other hierarchical

subcategories. For instance, Yahoo! "Outdoor" has only one common Level 2 subcategory in Google "Movie", and no common Level 2 subcategories in Google "Autos", "Photo", and "Software". Prior analysis reveals that the hierarchical structure of the source catalog in the real-world experimental data is different from that of the destination catalog.

4.2 Measurement

Since some documents may appear in more than one category of the same catalog, the number of test documents may vary slightly between Yahoo! and Google. This experiment followed an assumption in Agrawal and Srikant [2001] by measuring the performance of hierarchical catalog integration with accuracy defined in the following equation.

$$\frac{\text{Number of the test documents correctly integrated into } Di}{\text{Total number of the test documents in the dataset}} \tag{2}$$

To measure the performance of hierarchical catalog integration, Equation (2) was adopted in each level of the destination categories to assess its accuracy performance. In each level of the destination categories, the numerator denotes the test documents correctly integrated into that level, and the denominator represents the total test documents to be correctly integrated. The accuracy of each level in the destination categories and the average accuracy of the five categories were measured.

4.3 Results and Discussion

In the experiment, the documents were integrated both from Yahoo! into Google and from Google to Yahoo!. In the EHCI approach, the conceptual relationships between the hierarchical thesauri in both the source and destination categories added to an increasing λ value in the range 0–1. To further verify the effectiveness of the EHCI approach, three sets of negative examples were randomly chosen for Google training and the other three sets of negative examples were used for Yahoo! training. The overall performance of the EHCI approach is significantly boosted in all of these six sets. The best average performance improvements from Yahoo! to Google with the three sets of negative examples were 9.0%, 11.1%, and 21.6%. In contrast, the best performance improvements from Google to Yahoo! with the other three sets of negative examples were 18.1%, 21.6%, and 23.7%. Table 6 and Table 7, notably, list the medians and detail the average integration results with λ increasing from 0 to 1.

Table 6 shows the average catalog integration performance from Yahoo! to Google, and Table 7 lists that from Google to Yahoo!. Both first columns represent the λ values of the source catalog, and the first rows represent the λ values of the destination catalog. As indicated in Tables 6 and 7, the accuracy with the SHCI approach ($\lambda = 0.00$) from Yahoo! to

Google was 61.4% and that from Google to Yahoo! was 63.7%. The best performance improvements with EHCI were achieved at $\lambda = 0.01$ in the destination catalog and $\lambda = 0.30$ in the source catalog. The average accuracy from Yahoo! to Google and Google to Yahoo! was 72.5% and 85.3%, respectively.

Table 6. The average integration performance from Yahoo! to Google

S \ D	0.00	0.01	0.05	0.10	0.30	0.50	0.70	0.90	1.00
0.00	61.4%	61.1%	52.4%	38.4%	17.8%	15.1%	14.2%	13.9%	13.7%
0.01	60.7%	62.3%	54.9%	40.7%	18.4%	15.2%	14.3%	14.0%	13.7%
0.05	63.6%	66.2%	63.4%	52.1%	21.2%	15.9%	14.6%	14.2%	14.0%
0.10	66.3%	69.6%	68.0%	60.3%	27.6%	17.6%	15.3%	14.3%	14.2%
0.30	68.7%	**72.5%**	72.1%	68.5%	54.7%	35.9%	26.2%	18.2%	17.4%
0.50	67.1%	71.2%	71.7%	69.9%	61.6%	53.7%	40.1%	30.8%	28.0%
0.70	64.6%	68.5%	69.1%	68.2%	61.2%	58.7%	52.2%	41.3%	37.9%
0.90	64.1%	67.8%	69.0%	68.5%	62.7%	60.5%	56.8%	51.9%	47.6%
1.00	63.6%	67.4%	68.9%	68.5%	63.3%	61.5%	58.5%	53.8%	51.8%

Table 7. The average integration performance from to Google to Yahoo!

S \ D	0.00	0.01	0.05	0.10	0.30	0.50	0.70	0.90	1.00
0.00	63.7%	61.7%	33.1%	15.0%	0.8%	0.2%	0.1%	0.1%	0.1%
0.01	66.0%	65.6%	37.8%	16.9%	1.0%	0.2%	0.1%	0.2%	0.1%
0.05	72.4%	74.2%	54.2%	29.4%	1.8%	0.3%	0.2%	0.2%	0.1%
0.10	76.7%	80.4%	64.9%	42.2%	4.6%	0.6%	0.2%	0.2%	0.2%
0.30	81.8%	**85.3%**	75.5%	57.1%	29.7%	12.2%	2.3%	0.3%	0.2%
0.50	80.2%	85.0%	77.7%	60.4%	39.6%	26.2%	16.4%	8.4%	4.3%
0.70	77.5%	83.5%	77.8%	61.7%	43.2%	32.7%	24.3%	19.6%	14.5%
0.90	75.8%	82.5%	78.1%	62.9%	45.9%	38.5%	30.0%	24.2%	21.0%
1.00	75.2%	81.7%	78.3%	63.1%	46.6%	38.9%	32.0%	26.2%	24.2%

Since the best accuracy in both Google-to-Yahoo! and Yahoo!-to-Google integration was obtained by adding the hierarchical label weights with $\lambda = 0.30$, we can infer that the conceptual thesaurus extracted from the source hierarchy significantly improves hierarchical catalog integration. Conversely, the conceptual thesaurus extracted from the destination hierarchy to enhance the hierarchical classifiers is not as effective as the source hierarchical thesaurus. Tables 6 and 7 show that the improvement in accuracy obtained by changing from $\lambda = 0.00$ to $\lambda = 0.01$ was less than 5%. Experimental results indicate that the conceptual relationships in the source hierarchical thesaurus are more likely to enhance hierarchal Web catalog integration than those in the destination hierarchical thesaurus.

Tables 8 and 9 further describe the integration accuracy of the six hierarchical levels with $\lambda = 0.01$ in the destination catalog and $\lambda = 0.30$ in the source catalog. Analytical results indicate that the EHCI approach consistently improves the accuracy performance of each level in almost all cases. However, Table 8 still indicates that the EHCI approach induced a 2.7% accuracy decrease at Level 3 and a 23.8% accuracy decrease at Level 6 in the Software category when integrating from Yahoo! to Google. Table 9 also indicates a 0.7% accuracy decrease at Level 6 in the Movies category, and a 5.7% accuracy decrease at Level 6 in the Software category when integrating from Google to Yahoo!. The main reason for these falls in accuracy is probably due to the training documents in those destination levels lacking the hierarchical thesauri extracted from the source catalog.

Table 8. The Yahoo!-to-Google integration performance in six levels

	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	Total
Autos	0.0% (0)	84.9% (73)	50.0% (109)	40.5% (45)	52.6% (10)	0.0% (0)	54.2% (237)
Autos_E	0.0% (0)	91.9% (79)	74.8% (163)	60.4% (67)	68.4% (13)	66.7% (2)	74.1% (324)
Movies	0.0% (0)	63.4% (83)	75.4% (395)	56.6% (197)	54.0% (163)	37.1% (13)	63.5% (851)
Movies_E	0.0% (0)	71.8% (94)	80.0% (419)	61.5% (214)	70.2% (212)	40.0% (14)	71.1% (953)
Outdoors	0.0% (0)	70.0% (28)	75.0% (57)	69.6% (48)	79.2% (19)	46.7% (7)	71.0% (159)
Outdoors_E	0.0% (0)	72.5% (29)	93.4% (71)	87.0% (60)	87.5% (21)	73.3% (11)	85.7% (192)
Photo	0.0% (0)	42.3% (11)	55.7% (49)	39.0% (23)	36.0% (9)	25.0% (2)	45.6% (94)
Photo_E	0.0% (0)	61.5% (16)	68.2% (60)	57.6% (34)	52.0% (13)	25.0% (2)	60.7% (125)
Software	100.0% (2)	72.4% (21)	63.1% (94)	68.0% (164)	58.0% (91)	58.1% (61)	63.4% (433)
Software_E	100.0% (2)	86.2% (25)	60.4% (90)	82.6% (199)	73.2% (115)	34.3% (36)	68.4% (467)

Table 9. The Google-to-Yahoo! integration performance in six levels

	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6	Total
Autos	63.6% (7)	68.3% (86)	64.7% (77)	52.5% (53)	47.4% (18)	88.9% (8)	61.6% (249)
Autos_E	100.0% (11)	84.9% (107)	89.1% (106)	88.1% (89)	84.2% (32)	100.0% (9)	87.6% (354)
Movies	0.0% (0)	72.1% (101)	56.7% (102)	50.1% (177)	53.7% (217)	50.7% (68)	54.9% (665)
Movies_E	0.0% (0)	91.4% (128)	78.9% (142)	94.9% (335)	65.6% (265)	50.0% (67)	77.4% (937)
Outdoors	0.0% (0)	70.5%(31)	80.7% (92)	44.4% (8)	0.0% (0)	100.0% (1)	74.6% (132)
Outdoors_E	0.0% (0)	100.0% (44)	97.4% (111)	88.9% (16)	0.0% (0)	100.0% (1)	97.2% (172)
Photo	0.0% (0)	63.9% (46)	60.3% (47)	84.2% (16)	40.0% (2)	0.0% (0)	63.4% (111)
Photo_E	0.0% (0)	90.3% (65)	93.6% (73)	84.2% (16)	60.0% (3)	0.0% (0)	89.7% (157)
Software	100.0% (3)	83.6% (122)	75.9% (101)	76.1% (118)	79.9% (139)	71.4% (25)	78.6% (508)
Software_E	100.0% (3)	93.8% (137)	85.0% (113)	91.6% (142)	97.1% (169)	65.7% (23)	90.9% (587)

Figures 5 and 6 depict the overall performance between the EHCI and SHCI approaches. The results indicate that EHCI outperforms SHCI in both Yahoo!-to-Google and Google-to-Yahoo! catalog integration. Figure 5 indicates that the EHCI approach achieved an average accuracy improvement of 11.1% in Yahoo!-to-Google catalog integration. In Figure 6, the EHCI approach obtained an average accuracy improvement of 21.6% in Google-to-Yahoo! catalog integration. The results further indicate that hierarchical catalog integration can be effectively boosted by enhancement of the conceptual relationships extracted from the hierarchical thesauri.

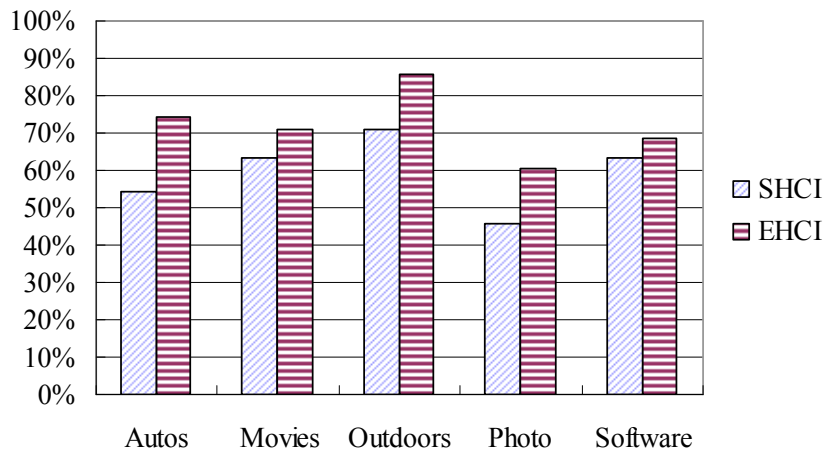

Figure 5. The average integration performance from Yahoo! to Google

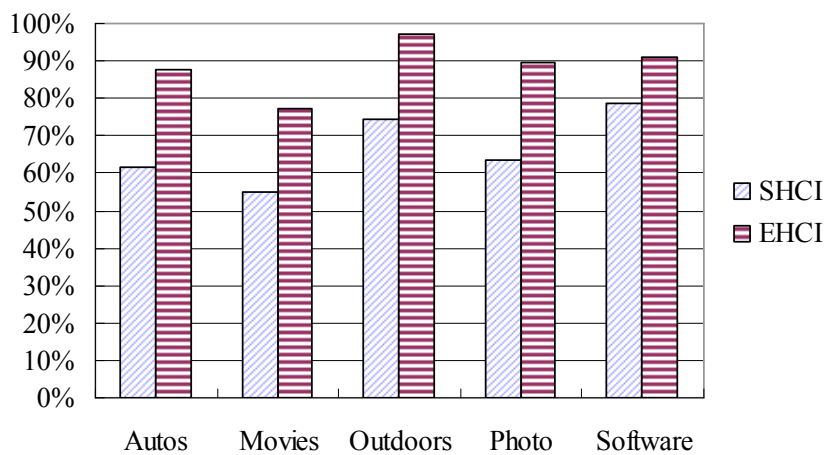

Figure 6. The average integration performance from Google to Yahoo!

As well as the accuracy performance, the computation cost of SHCI and EHCI approaches was further analyzed. The experimental environment was in an IBM PC with an Intel Core Duo T2400 CPU and 1GB memory. The overall CPU runtime provided by SVM^{light} was to analyze the training and testing time, excluding the data I/O time, in a Windows XP environment. Results of runtime analysis demonstrate that SHCI took 65.60 seconds to perform Google-to-Yahoo! catalog integration and 6.40 seconds to perform Yahoo!-to-Google catalog integration. Conversely, EHCI took 86.53 seconds to perform Google-to-Yahoo! catalog integration and 6.69 seconds to perform Yahoo!-to-Google catalog integration. The reason for the faster CPU time of Yahoo!-to-Google integration is the much smaller number of Google classifiers than Yahoo! classifiers. The CPU runtime analysis further indicates that the proposed approach can efficiently complete the catalog integration work.

5. Conclusion

Web catalog integration is a significant issue in Web content management. Although past studies have indicated that a hierarchical structure is superior to a flattened structure in classification, recent studies have only presented a few primitive results and have not comprehensively studied hierarchical structures in hierarchical Web catalog integration. This study addresses the problem of hierarchical catalog integration, and proposes an enhanced hierarchical catalog integration (EHCI) scheme.

This study further reports experimental results concerning the improvement in Web catalog integration accuracy resulting from the use of EHCI. The integration accuracy is significantly improved by exploiting the conceptual relationships extracted from the source and destination catalog thesauri to enhance hierarchical catalog integration. Experimental results indicate that EHCI is effective for hierarchical Web catalog integration, and achieves improvements in almost every hierarchical level on real-world catalogs with SVM classifiers. In overall performance of hierarchical catalog integration, the EHCI approach can consistently improve accuracy in real-world catalog integration.

To conclude, this study demonstrates that the conceptual relationships learned from the source and destination catalog thesauri can enhance hierarchical catalog integration. Experimental results indicate that the accuracy improvements in a hierarchical structure are very promising, especially the hierarchical thesaurus extracted from the source catalog. Future work will involve investigating other classification models in order to build an integration platform for hierarchical catalog integration. Furthermore, complex catalog integration issues will be considered through ontology relationships.

Acknowledgement

The authors would like to thank the National Science Council of the Republic of China, Taiwan, for partially supporting this research under Contract No. NSC 95-2745-E-155-008. The authors would also like to express many thanks to the anonymous reviewers for their precious suggestions.

References

Agrawal, R., and R. Srikant., "On Integrating Catalogs," in *Proceedings of the 10th WWW Conf. (WWW10)*, Hong Kong, 2001, pp. 603–612.

Chen, I.-X., C.-Z. Yang, and J.-C. Ho, "An Iterative Approach for Web Catalog Integration with Support Vector Machines," in *Proceedings of Asia Information Retrieval Symposium 2005 (AIRS2005)*, Jeju Island, Korea, 2005, pp. 703–708.

Chen, I.-X., C.-Z. Yang, and J.-C., Ho, "On Hierarchical Web Catalog Integration with Conceptual Relationships in Thesaurus," in *Proceedings of the 29th Annual ACM Conf. on Research and Development in Information Retrieval (SIGIR'06)*, Settle, Washington, USA, 2006, pp. 635–636.

Doan, A., J. Madhavan, P. Domingos, and A. Halevy, "Learning to Map between Ontologies on the Semantic Web," in *Proceedings of the 11th WWW Conf. (WWW2002)*, Honolulu, Hawaii, 2002, pp. 662–673.

Dumais, S., J. Platt, D. Heckerman, and M. Sahami, "Inductive Learning Algorithms and Representations for Text Categorization," in *Proceedings of the 7th Int'l Conf. on Information and Knowledge Managemen (CIKM)*, Bethesda, Maryland, USA, 1998, pp. 148–155.

Dumais, S., and H. Chen, "Hierarchical Classification of Web Content," in *Proceedings of the 23rd Annual ACM Conf. on Research and Development in Information Retrieval (SIGIR'00)*, Athens, Greece, 2000, pp. 256–263.

Frakes, W., and R. Baeza-Yates, *Information Retrieval: Data Structures and Algorithms*, Prentice Hall Press, USA, 1992.

Foskett, D.J., "Thesaurus," in *Readings in Information Retrieval*, Jones, K.S., and Willett, P., Ed. Morgan Kaufmann Press, San Francisco, CA, USA, 1997, pp. 111 –134.

Ho, J.-C., I.-X. Chen, and C.-Z. Yang, "Learning to Integrate Web Catalogs with Conceptual Relationships in Hierarchical Thesaurus," in *Proceedings of Asia Information Retrieval Symposium 2006 (AIRS2006)*, Singapore, 2006, pp. 217–229.

Joachims, T., "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," in *Proceedings of the 10th European Conf. on Machine Learning (ECML'98)*, Chemnitz, DE, 1998, pp. 137–142.

Keller, A. M., "Smart Catalogs and Virtual Catalogs," in *Readings in Electronic Commerce*, Kalakota, R., and Whinston, A., Ed. Addison-Wesley Press, USA, 1997.

Kim, D., J. Kim, and S. Lee, "Catalog Integration for Electronic Commerce through Category-Hierarchy Merging Technique," in *Proceedings of the 12th Int'l Workshop on Research Issues in Data Engineering: Engineering e-Commerce/e-Business Systems (RIDE'02)*, San Jose, CA, USA, 2002, pp. 28–33.

MacCallum, A., R. Rosenfeld, T. Mitchell, and A. Ng, "Improving Text Classification by Shrinkage in a Hierarchy of Classes," in *Proceedings of the 15th International Conference on Machine Learning (ICML-98),* Madison, Wisconsin, 1998, pp. 359–367.

Marrón, P. J., G. Lausen, and M. Weber, "Catalog Integration Made Easy," in *Proceedings of the 19th Int'l Conf. on Data Engineering (ICDE'03)*, Bangalore, India, 2003, pp. 677–679.

Rajan, S., K. Punera, and J. Ghosh, "A Maximum Likelihood Framework for Integrating Taxonomies", *in Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05),* Pittsburgh, Pennsylvania, pp. 856–861.

Rennie, J. D. M., and R. Rifkin, "Improving Multiclass Text Classification with the Support Vector Machine," *Technical Report AI Memo AIM-2001-026 and CCL Memo 210*, MIT Press, USA, 2001.

Sarawagi, S., S. Chakrabarti, and S. Godbole, "Cross-Training: Learning Probabilistic Mappings between Topics," in *Proceedings of the 9th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, Washington, D.C., 2003, pp. 177–186.

Stonebraker, M., and J. M. Hellerstein, "Content Integration for e-Commerce," in *Proceedings of the 2001 ACM SIGMOD Int'l Conf. on Management of Data*, Santa Barbara, CA, USA, 2001, pp. 552–560.

Sun, A., and E.-P. Lim, "Hierarchical Text Classification and Evaluation," in *Proceedings of the 2001 IEEE Int'l Conf. on Data Mining (ICDM'01)*, Washington, D.C., USA, 2001, pp. 521–528.

Sun, A., E.-P. Lim, and W.-K. Ng, "Performance Measurement Framework for Hierarchical Text Classification," *Journal of the American Society for Information Science and Technology (JASIST)*, 54(11), 2003, pp. 1014–1028.

Tsay, J.-J., H.-Y. Chen, C.-F. Chang, and C.-H. Lin, "Enhancing Techniques for Efficient Topic Hierarchy Integration," in *Proceedings of the 3rd Int'l Conf. on Data Mining (ICDM'03)*, Melbourne, FL, USA, 2003, pp. 657–660.

Vural, V., and J. G. Dy, "A Hierarchical method for Multi-Class Support Vector Machine," in *Proceedings of the Int'l Conf. on Machine Learning 2004 (ICML2004)*, Banff, Alberta, Canada, 2004, pp. 105.

Wu, C.-W., T.-H. Tsai, and W.-L. Hsu, "Learning to Integrate Web Taxonomies with Fine-Grained Relations: A Case Study Using Maximum Entropy Model," in *Proceedings of Asia Information Retrieval Symposium 2005 (AIRS2005)*, Jeju Island, Korea, 2005, pp. 190–205.

Yang, Y., and X. Liu, "A Re-examination of Text Categorization Methods," in *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval (SIGIR'99)*, Berkeley, CA, USA, 1999, pp. 42–49.

Zhang, D., and W.S. Lee, "Web Taxonomy Integration using Support Vector Machines," in *Proceedings of the 13th WWW Conf. (WWW2004),* New York, NY, USA, 2004a, pp. 472–481.

Zhang, D., and W.S. Lee, "Web Taxonomy Integration through Co-Bootstrapping," in *Proceedings of the 27rd Annual ACM Conf. on Research and Development in Information Retrieval (SIGIR'04)*, Athens, Greece, Sheffield, United Kingdom, 2004b, pp. 410–417.

Online Resources

Google Catalog: http://www.google.com/dirhp

Joachims, T., SVM*light*:, version 5.0, http://svmlight.joachims.org/, 2002

Porter, M., Porter Stemmer: http://www.tartarus.org/martin/PorterStemmer/, 1980

Yahoo! Catalog: http://dir.yahoo.com/

Wikipedia http://en.wikipedia.org/wiki/Thesaurus