Web Catalog Integration using Support Vector Machines

Ing-Xiang Chen Chih-Hsing Shih Cheng-Zen Yang
Department of Computer Science and Engineering
Yuan Ze University
Chungli, Taiwan, R.O.C.
{sean,chihsing,czyang}@syslab.cse.yzu.edu.tw

摘要

目錄整合在電子商務領域中式個重要的研究議題。過往的研究曾經顯示出，如果能夠利用來源目錄所隱含的目錄資訊，整合的精確性將會被有效的改善。然而在過往針對支援向量機（SVM）的研究中，若不是沒有顯著的改進，就是只能改進部分情況而無法全面改善。因此，本論文針對三種不同SVM的整合效能加以討論，並且利用來源目錄隱含資訊，加以同義詞擴展以及特徵擴展的機制，進一步提升SVM的整合效能。我們用真實的Web目錄加以測試，實驗結果顯示SVM優於以往研究中的NB分類法，同時我們所提出的改進方法能全面性地提升SVM整合效能。

關鍵詞：目錄整合，分類，支援向量機，同義詞擴展，特徵擴展。

Abstract

Catalog integration has been addressed as an important issue in e-catalog management. In the past studies, it has been shown that exploiting the implicit information embedded in the source catalog can highly improve the integration accuracy. However, previous enhancement approaches based on support vector machines (SVM) can achieve just slightly improvements or in nearly half the cases. In this paper, we report our studies on three different SVM classifiers, and propose an enhancement approach to improve SVM with implicit source information, synonym expansion, and feature expansion. We have conducted several experiments using real-world Web catalogs. The experimental results justify that SVM outperforms NB-based approaches. It is also shown that our approach consistently achieves improvements on SVM classifiers.

Keywords: Catalog Integration, Classification, Support Vector Machines, Synonym Expansion, Feature Expansion.

1. Introduction

As the environment of e-commerce is rapidly growing, catalog integration has been addressed as an important issue in e-catalog management [1,10,11,14]. According to the taxonomy in [14], corporate Web usage can be classified into three generations, and catalog integration is crucial to the current model, the third generation e-business model. To provide online content-rich information, distributors need to integrate the catalogs from each supplier. However, to the best of our knowledge, there is no *de facto* catalog standard for systematical catalog integration. Despite that some catalog models such as Virtual Catalogs [9] and Live Catalog [10] have been proposed, they are still in the laboratory research stage. The absence of robust systematic models implies that directly manipulating the original catalog content is not avoidable, and thus catalog integration in turn comprises a lot of classification work.

The simplest way to perform catalog integration would be just to re-classify the items in the supplier's catalog (the source catalog, *S*) into the distributor's catalog (the destination catalog, *D*). However, without exploiting the implicit information embedded in the source catalog, the classification accuracy is limited to the classifier [1,16]. As reported in [1], a standard Naive Bayes (NB) approach without exploiting implicit category information can achieve at most 65.2% accuracy while merging the Yahoo! categorization into the corresponding Google categorization. On the contrast, their enhanced NB approach (NB-AS) can achieve 77.8% accuracy while merging the same category. In [16], similar results are also reported.

Although it is shown that these enhancements on NB classifiers can highly improve the integration accuracy, few research efforts [13,16] have been concentrated on Support Vector Machines (SVM) [18], a superior classification approach reported in past literature [4,7,12,19]. In [13], a cross-trained SVM (SVM-CT) is proposed to merge two catalogs. However, SVM-CT outperforms SVM in only nearly half the cases. Meanwhile, [16] uses an approach called *topic restriction* in NB and SVM to reduce the inaccuracy by restricting the classification of any document to a small set of candidate destination categories. A candidate category is decided if more than a predefined number of common documents appear in both source and destination categories. Although this approach can significantly improve the performance of NB, it can only slightly improve the performance of SVM.

In this paper, we study the effectiveness of SVM classifiers in catalog integration by using embedded implicit catalog information and auxiliary synonym information of Web documents. The synonym information is considered in the integration phase to help alleviate the vocabulary change problem mentioned in [1]. We follow the basic idea mentioned in [1]: if two documents belong to the same category in *S*, they are more likely to belong to the same category in *D*. The likeliness is then decided from the following process: training SVM with *D*; using SVM to first classify the source documents into *D*; and expanding the SVM training set by including the auxiliary information of newly classified documents. Two issues are further studied in this paper. First, since SVM can use different kernel functions, we in advance study SVMs with three popular learning functions, linear SVM, polynomial SVM (poly-SVM) and Gaussian radial basis function (RBF-SVM), to investigate which learning function best performs in category integration. Second, since every different part of a document may have different discrimination power, text weighting is also studied.

We have conducted several experiments using real-world catalogs from two well-known search engines, Yahoo! and Google, to study the performance of SVMs and the enhancement of training set expanding. In all experiments, SVM^{light} [8] was used for all classification tasks. The experimental results show that all SVMs outperform NB in all cases, and poly-SVM outperforms linear SVM and RBF-SVM in most cases. This is consistent with the previous observations that SVMs generally beat NB in text classification. With the help of training set expanding, the accuracy of all SVMs is significantly improved. It has been shown that our approach consistently achieves improvements on SVM classifiers.

The rest of the paper is organized as follows. Section 2 first states the problem definitions, assumptions, and limitations. Section 3 reviews the past research work on catalog integration. In Section 4, we describe the details of training set expanding, and discuss several design issues. Section 5 presents our experimental evaluation of accuracy and discusses the factors that influence the experiments. Finally, Section 6 gives a concluding remark and discusses some future research directions.

2. Problem Statement and Terminology

Like the formal definitions in [1], we assume that there are two catalogs participating in the integration process. One is the source catalog *S* with a set of *m* categories S_1, S_2,…,S_m. Another is the destination catalog *D* with a set of n categories $D_1, D_2,...,D_n$. The integration process is performed by merging each document *d* in *S* into a correspondent category in *D*. Optimistically, if a source category S_i has similar attributes with a destination category D_j, each document *d* in S_i should be able to be merged into D_j. However, this “winner-takes-all” attitude neglects the connotative differences of two catalogs in their organization principles. At another extreme end, all the attributes of the source categories can be discarded and each document *d* in the source catalog is just reclassified into the destination catalog. However, this straightforward reclassification approach ignores the implicit valuable information contained in the source categories. Therefore, the catalog integration problem can be viewed as to merge every *d* in *S* into *D* with the assistance of the implicit information of *S*.

To simplify the study on SVM’s performance, we follow the integration model used in [1] in which the category hierarchies are flattened. Although this cannot model many real-world cases in which catalogs are hierarchical, the flat catalog assumption is still helpful in investigating the effectiveness of incorporating the implicit source information in catalog integration. Since past studies [2,3] have shown that using hierarchical information can further improve the classification performance, generalizing the flat catalog assumption to the hierarchical catalog model is left for our future research. In addition to this flat catalog assumption, we also assume that the catalogs are homogeneous and overlapped with some common documents. This means that the catalogs are not orthogonal, so the implicit source information can be exploited. Our real-world data sets also support this overlapping assumption.

To classify source documents, the SVM classifier needs to be trained first. In the training process, all training documents come from the destination catalog. Whether a training document is treated as a positive document or a negative document is subject to its subordinate relationship to each destination category. Thus, the SVM classifier is a “one-against-all” classifier.

In our study, three well-known kernel functions were used in the SVM classifier. A linear SVM uses a real-valued function $f : X \in R^n \to R$ to find 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 of $f(x) = \langle w, x\rangle + b = \sum_{i=1}^{n} w_i x_i + b$ where $(w,\ b) \in R^n \times R$. The linear SVM is trained to find the optimal values of *w* and *b* such that $\|w\|$ is minimized.

In a polynomial kernel, the decision boundary of the hyperplane is determined by a polynomial curve of degree *d*, $K(x_i, x_j) = \langle x_i, x_j \rangle^d$. In a Guassian radial basis kernel, the decision boundary is a Guassian function, $K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$. Then the SVM classifier is defined as $f(x) = \sum_{i,j=1}^{l} \alpha_i y_i K(x_i x_j) + b$ in the Hilber space.

In the training phase, the classifiers try to find the optimal values of α_i and b, and decide the optimal decision boundary for each separation hyperplane. These trained classifiers are then used in the following integration process.

3. Related Research

The past integration research can be classified into two main research fields according from different integration perspectives. One main research field is to discuss the integration methodology from a viewpoint of integration architecture [10,11,14]. In [14], several integration challenges such as query processing and content mapping are discussed. In [10,11], different architectures are proposed to facilitate catalog integration by using category rules and integration fitness functions. However, the rule-based integration proposed in [10] heavily relies on a manually defined catalog graph describing the metadata of the catalog. In many integration cases, such information is actually unavailable for the integrator. On the other hand, though the fitness functions proposed in [11] can achieve very low integration error-rates, they are query-dependent. Besides, the integration performance of the fitness functions depends on whether the catalogs are stored in XML format. Thus, the XML-dependency limits the applicability of the fitness functions.

Another main research filed is to exploit the power of text classifiers in catalog integration. In [1], an enhanced Naive Bayes approach (NB-AS) is proposed to improve the integration accuracy by exploiting implicit category information. In the experiments with real Web catalogs, NB-AS can achieve 30% fewer errors on average. The promising results show that exploiting implicit category information indeed benefits the accuracy of automatic catalog integration. However, they studied only the NB classifiers.

In [16], Tsay et al. have proposed two techniques to improve the accuracy of classifiers. The first technique is called *probabilistic enhancement* (PE) that uses category information to enhance probabilistic classifiers such as NB classifiers. The second technique is called *topic restriction* (TR) that can be applied to general classifiers such as SVM. The experimental results show that both techniques can significantly improve the accuracy of NB classifiers. For SVM, however, the TR enhancement can only achieve less than 0.2% improvement.

In [13], a *cross-training* (CT) technique has been proposed for NB and SVM to improve the integration accuracy by exploiting the native category information of *half-labeled* documents, which are assigned to only one catalog. In the CT phase, two half-labeled document sets are crossly used as the sample sets to train two classifiers. The experimental results show that the CT technique can achieve significant improvements for NB classifiers in most cases. However, SVM-CT outperforms SVM in only nearly half the cases.

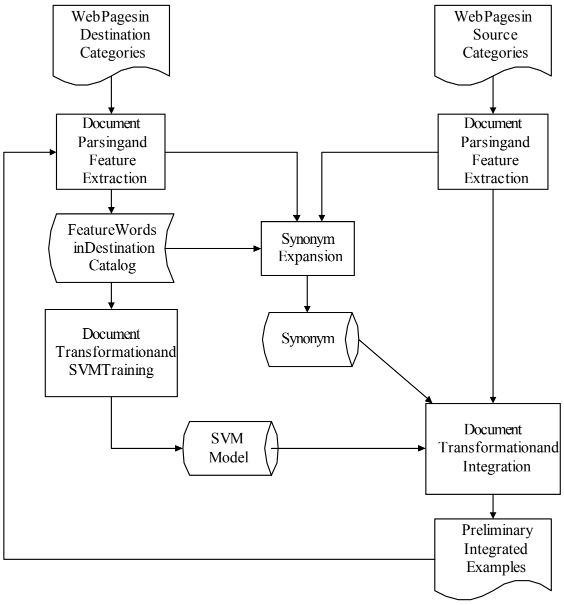
WebPagesin Destination Categories
WebPagesin Source Categories
Document Parsingand Feature Extraction
Document Parsingand Feature Extraction
FeatureWords inDestination Catalog
Synonym Expansion
Document Transformationand SVMTraining
Synonym
SVM Model
Document Transformationand Integration
Preliminary Integrated Examples

Figure 1: The integration process using SVM.

4. SVM with Auxiliary Information Enhancements

In catalog integration, the flattened source catalog S with a set of m categories $S_1, S_2, ..., S_m$ is intended to be merged into the flattened destination catalog D with a set of n categories $D_1, D_2, ..., D_n$. Since a binary SVM can solve only two-class classification problems, we adopt a "one-against-all" strategy to decompose a multi-class problem into a set of binary SVM problems. Positive and negative training data are respectively composed of the feature information extracted from the destination class and other non-destination classes as in [20]. A set of binary SVM classifiers then are trained for the integration process of each destination category.

4.1 Integration Process

Figure 1 shows the catalog integration process. The set of documents in the destination catalog is parsed first to extract the feature words as the training input of the SVM. In feature extraction, the stopwords are

Table 1: The 5-level weight assignment for hypertext processing.

Hypertext	Weight
Plain Text	1
H3 Header	3
H2 Header	5
H1 Header	7
Anchor Text	7
Title	10

Table 2: The categories used in the experiments.

Category	Yahoo! (dir.yahoo.com)	Docs	Google (directory.google.com)	Docs	Common Docs
Autos	Recreation_Sports/Autos/	18162	Shopping/Autos/	7297	498
Outdoors	Recreation_Sports/Outdoors/	14155	Recreation/Outdoors/	12755	1113
Software	Computers_and_Internet/Software/	14288	Computers/Software/	27587	1466
Movies	Entertainment/Movies_and_Film	2667	Top/Arts/Movies	1591	134
Photography	Arts/Visual_Arts/Photography	6005	Top/Arts/Photography	2797	327

removed, and the remaining feature words are weighted according to the surrounding HTML tags. Now we use a 5-level weight assignment to reflect the discriminative power of each feature word. The weighting approach has been shown very effective in many Web classification papers [6,15]. Table 1 shows the weight assignments currently used in our research. We have experimented with different weights and found that the title weighting has the significant impact. The score of a feature word is calculated by the following equation: $Score = p \times 1 + h3 \times 3 + h2 \times 5 + h1 \times 7 + a \times 7 + t \times 10$ where *p, h1, h2, h3, a*, and *t* are the numbers of occurrences.

4.2 Synonym Expansion

To cope with the vocabulary change problem, synonym information is also expended into the feature space. However, to avoid the negative effect of incorporating misleading polysemy words in synonym expansion, synonyms are decided according to the following procedure.

Suppose that the source category is S_k and the destination category is D_i. First, to the destination category D_i, we get a destination synonym set DS_i. Next we perform synonym filtering for DS_i to remove the words appearing in the other synonym sets DS_j, $j \neq i$, and in the feature word set of the destination catalog FW_{Di} to avoid introducing misleading polysemy words. After synonym filtering, a disjoint synonym set T_i is generated to represent the synonym information of the destination category D_i, and to be used in the integration phase.

In the integration phase, the synonym information of the source catalog is also incorporated to enhance the discriminative power of the classifier. We perform synonym expansion on the feature words of the source category S_k excluding the words appearing in the destination category D_i, and get a source synonym set SS_k. However, only a part of words in SS_k should be used in the integration process because if a synonym does not appear in FW_{Di} or T_i, it is highly possible to result in misclassification. Thus, only synonyms in $SS_k \cap FW_{Di}$ or $SS_k \cap T_i$ are used in the integration phase.

4.3 Feature Expansion

In the integration phase, the feature words of the source documents that have been integrated are incorporated as the implicit catalog information to re-train the SVM classifiers. There are two thresholds to control the number of expanded feature words. One is the term frequency, the number of term occurrences in the integrated source documents. Another is document frequency, the number of documents in which the term appears. Since if two documents belong to the same category in *S*, they are more likely to belong to the same category in *D*, and the newly expanded feature words will be beneficial to catalog integration.

5. Experiments

We have conducted several experiments using real-world catalogs from two well-known search engines, Yahoo! and Google, to study the performance of SVMs and the enhancement of training set expanding. In all experiments SVM$^{\text{light}}$ [8] was used for all classification tasks. The experimental results show that all SVMs outperform NB in all cases, and poly-SVM outperforms linear SVM and RBF-SVM in most cases.

5.1 Data Sets

Five categories from Yahoo! and Google were extracted in our experiments. Table 2 shows these categories and the number of the extracted documents after ignoring the documents that could not be extracted. As in [1,13], the documents appearing in only one category were used as the destination catalog *D*, and the common documents were used as the source catalog *S*. Thus we measured the accuracy by $\dfrac{\text{Number of correctly classified docs in } S}{\text{Total number of docs in } S}$. In the processing, we used the stopword list in [5] to remove the stopwords. Synonym expansion was based on the WordNet 2.0 thesauri [17].

5.2 Experimental Settings

In our experiments, NB and NB-AS classifiers were implemented for comparison. In the NB classifiers, the posterior probability of destination

Table 3: The accuracy of catalog integration from Google into Yahoo!.

	NB	NB-AS	Linear	Linear SVM	Poly-SVM	RBF-SVM
			SVM	+HW+SE+FE	+HW+SE+FE	+HW+SE+FE
Autos	214	252	379	397	400	402
498	42.97%	50.60%	76.10%	79.72%	80.32%	80.72%
	(λ=0.9)	(λ=0.9)				
Outdoors	460	525	915	944	947	918
1113	41.32%	47.16%	82.21%	84.82%	85.09%	82.48%
	(λ=0.9)	(λ=0.9)				
Software	467	546	1258	1272	1276	1244
1466	31.85%	37.24%	85.81%	86.77%	87.04%	84.86%
	(λ=0.9)	(λ=0.9)				
Movies	34	38	98	98	99	91
134	25.37%	28.35%	73.13%	73.13%	73.88%	67.91%
	(λ=0.1)	(λ=0.1)				
Photography	186	218	262	266	268	248
327	56.88%	66.66%	80.12%	81.35%	81.96%	75.84%
	(λ=0.9)	(λ=0.9)				

category D_i for the documents of source catalog is estimated as in [1]. For each document *d* in *S*, the posterior probability of destination category D_i is $\Pr(D_i|d)=\frac{\Pr(D_i)\Pr(d|D_i)}{\Pr(d)}$, where Pr($D_i$) is estimated by the equation $\frac{\text{Number of documents in } D_i}{\text{Total number of docs}}$, $\Pr(d|D_i)=\prod_{t\in d}\Pr(t|D_i)$, and *t* is the words. Pr($t|D_i$) then can be simply estimated by the maximum likelihood estimate *num*(D_i, *t*)/*num*(D_i), where *num*(D_i, *t*) is the number of occurrences of t in category D_i, and *num*(D_i) is the total number of words in category D_i. Since this estimation may be zero, it is smoothed by using the Lidstone's smoothing parameter λ as in [1, 13]. Then Pr($t|D_i$) is smoothed by $\frac{num(D_i,t)+\lambda}{num(D_i)+\lambda|V|}$, where $0 \le \lambda \le 1$, and $|V|$ is the number of words in the vocabulary. In NB-AS, the posterior probability for document *d* is changed to $\Pr(D_i|d,S_j)=\frac{\Pr(D_i|S_j)\Pr(d|D_i)}{\Pr(d|S_j)}$, where S_j is a category in *S* and contains *d*.

5.3 Results

Table 3 lists the experimental results of integrating Google's pages into Yahoo's categories. Table 4 lists the experimental results of reversely integrating

Table 4: The accuracy of catalog integration from Yahoo! into Google.

	NB	NB-AS	Linear	Linear SVM	Poly-SVM	RBF-SVM
			SVM	+HW+SE+FE	+HW+SE+FE	+HW+SE+FE
Autos	239	268	407	415	415	397
498	47.99%	53.81%	81.37%	83.33%	83.33%	79.72%
	(λ=0.5)	(λ=0.5)				
Outdoors	494	544	972	983	989	964
1113	44.38%	48.87%	87.33%	88.32%	88.86%	86.61%
	(λ=1.0)	(λ=1.0)				
Software	468	520	1332	1341	1344	1347
1466	31.92%	35.47%	90.86%	91.47%	91.68%	91.88%
	(λ=1.0)	(λ=1.0)				
Movies	36	39	100	100	100	85
134	26.86%	29.10%	74.63%	74.63%	74.63%	63.43%
	(λ=0.2)	(λ=0.2)				
Photography	185	201	277	278	279	230
327	56.57%	61.46%	84.71%	85.02%	85.32%	70.34%
	(λ=0.3)	(λ=0.3)				

Table 5: The accuracy of catalog integration from Google into Yahoo!. Different weights are assigned to the tagged hypertext.

	Linear SVM No weighting	Linear SVM+HW (1,1,1,1,1,10)	Linear SVM+HW (1,3,5,7,7,10)
Autos 498	379 76.10%	395 79.32%	395 79.32%
Outdoors 1113	915 82.21%	937 84.19%	938 84.28%
Software 1466	1258 85.81%	1263 86.15%	1263 86.15%
Movies 134	98 73.13%	97 72.39%	97 72.39%
Photography 327	262 80.12%	261 79.82%	261 79.82%

Yahoo's pages into Google's categories. As listed in two tables, we have measured the accuracy achieved by the following classifiers: NB, NB-AS, linear SVM, enhanced linear SVM, enhanced poly-SVM, and enhanced RBF-SVM. In the tables, the optimized choices of λ values are also denoted for NB and NB-AS. The enhancements on SVM include hypertext weighting (HW), synonym expansion (SE), and feature expansion (FE).

Table 3 and Table 4 show that NB-AS outperforms NB, and all SVMs outperform NB-AS. The results are consistent to the previous studies in [1,4,7,12,19]. Meanwhile, with our enhancements, the accuracy of all SVMs is improved. Although the improvement ratios achieved by the enhanced SVMs (compared with linear SVM) are not as significant as the improvement ratios achieved by NB-AS (compared with NB), we can find that our approach consistently achieves improvements on SVM classifiers.

Table 5 studies the effects of different hypertext weighting. Here we only list the experimental results of integrating Google's pages into Yahoo's categories. From Table 5, we observe that assigning different weights to tagged text can improve the integration accuracy in three categories: `Auto`, `Outdoors`, and `Software`. In other two categories, the integration accuracy remains comparable. In addition, the title weighting has the significant impact.

From the experimental results, we can also observe that in the most cases the enhanced poly-SVM outperforms other SVMs and the enhanced RBF-SVM has the worst performance. These results show that our enhancements are more suitable for poly-SVM and linear SVM. However, we still cannot find a reason behind these results. One possible reason could be that the linear functions and polynomial functions are in a simpler form than the RBF functions.

6. Conclusions

Catalog integration is an important issue in current e-commerce applications. In this paper, we report our studies on support vector machines (SVM), and an approach for enhancing the integration accuracy. We compared our approach with the Naive Bayes classifier and the enhanced NB-AS classifier. The experimental results are very promising. All SVMs outperform NB and NB-AS in all cases, and poly-SVM outperforms linear SVM and RBF-SVM in most cases. This is consistent with the previous observations that SVMs generally beat NB in text classification. With the help of training set expanding, the accuracy of all SVMs is significantly improved. It has been shown that our approach consistently achieves improvements on SVM classifiers.

There are still many issues left for further discussion. First, generalizing the flat catalog assumption to the hierarchical catalog model is of the major interest for the catalog integration problem because hierarchical catalogs are more practical in real cases. Second, constructing a systematical mechanism to finding a better kernel function is a more difficult problem but can investigate the power of SVM. To conclude, we believe that the accuracy of catalog integration can be further improved with the assistance of more effective auxiliary information.

References

[1] R. Agrawal and R. Srikant. "On Integrating Catalogs." Proc. of the 10th WWW Conf. (WWW10), pp. 603-612, Hong Kong, May 2001.

[2] V. Boyapati. "Improving Hierarchical Text Classification Using Unlabeled Data." Proc. of the 25th Annual ACM Conf. on Research and Development in Information Retrieval (SIGIR'02), pp. 363-364, Tampere, Finland, Aug. 2002.

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

[4] S. Dumais, J. Platt, D. Heckerman, and M. Sahami. "Inductive Learning Algorithms and Representations for Text Categorization." Proc. of the 7th Int'l Conf. on Information and Knowledge Management, pp. 148-155. 1998.

[5] W. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*. Prentice Hall, PTR., 1992.

[6] C. Jenkins and D. Inman. "Adaptive Automatic Classification on the Web." Proc. of the 11th Int'l Workshop on Database and Expert Systems Applications, pp. 504-511, Greenwich, London, U.K., Sept. 2000.

[7] T. Joachims. "Text Categorization with Support

Vector Machines: Learning with Many Relevant Features." Proc. of the 10th European Conf. on Machine Learning (ECML'98), pp. 137-142, Chemnitz, DE, 1998.

[8] T. Joachims. "Making Large-Scale SVM Learning Practical." In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

[9] A. M. Keller. "Smart Catalogs and Virtual Catalogs." In Ravi Kalakota and Andrew Whinston, editors, *Readings in Electronic Commerce*. Addison-Wesley, 1997.

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

[11] P. J. Marrón, G. Lausen, and M. Weber. "Catalog Integration Made Easy." Proc. of the 19th Int'l Conf. on Data Engineering (ICDE'03), pp. 677-679, Bangalore, India, Mar. 2003.

[12] J. D. M. Rennie and R. Rifkin. "Improving Multiclass Text Classification with the Support Vector Machine." Tech. Report AI Memo AIM-2001-026 and CCL Memo 210, MIT, Oct. 2001.

[13] S. Sarawagi, S. Chakrabarti, and S. Godbole. "Cross-Training: Learning Probabilistic Mappings between Topics." Proc. of the 9th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, pp. 177-186, 2003.

[14] M. Stonebraker and J. M. Hellerstein. "Content Integration for e-Commerce." Proc. of the 2001 ACM SIGMOD Int'l Conf. on Management of Data, pp. 552-560, Santa Barbara, CA, May 2001.

[15] A. Sun, E.-P. Lin, and W.-K. Ng. "Web Classification Using Support Vector Machine." Proc. of the 4th Int'l Workshop on Web Information and Data Management, pp. 96-99, Nov. 2002.

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

[17] http://www.cogsci.princeton.edu/˜wu/wn2.0.

[18] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer Verlag, Heidelberg, DE, 1995.

[19] Y. Yang and X. Liu. "A Re-examination of Text Categorization Methods." Proc. of the 22nd Annual ACM Conference on Research and Development in Information Retrieval, pp. 42–49, Berkeley, CA, Aug. 1999.

[20] B. Zadrozny. "Reducing Multiclass to Binary by Coupling Probability Estimates." In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems* 14 (NIPS 2001), Cambridge, MA, 2002. MIT Press.