A Web Site Mining Algorithm Using the Multiscale Tree Representation Model

YongHong Tian[1], TieJun Huang[1,3],Wen Gao[1,2,3]
[1](Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China);
[2](Dept. of Computer Science, Harbin Institute of Technology, Harbin 150001, China)
[3](Graduate School of Chinese Academy of Sciences, Beijing 100039, China)
E-mail: {yhtian,tjhuang,wgao}@jdl.ac.cn

ABSTRACT

Web site mining, which aims at automatically discovering and classifying topic-specific web sites from the World Wide Web, has attracted increasing attention as indicated by the exponential growth of both the amount and the diversity of the web information. This paper describes a novel multiscale approach for web site mining, which represents a web site as a multiscale site tree, extending the existing tree representation models of web sites to an extra level of resolution (Document Object Model or DOM nodes). Furthermore, the hidden Markov tree (HMT) is utilized to model the intrascale contextual dependencies in the multiscale site tree, and a context-based fusion algorithm is applied to combining the interscale context models with the HMT-based classifiers in order to refine the raw classification results. Moreover, for further improving classification accuracy while reducing the classification overheads, we introduce a two-stage text-based denoising procedure to remove the "noise" information within web sites, and an entropy-based approach to dynamically prune the site trees. Experiments show that our approach achieves in average 16% improvement in classification accuracy and 34.5% reduction in processing time over the baseline system.

KEY WORDS: web site mining, multiscale site tree, context models, hidden Markov tree (HMT), interscale fusion, entropy-based pruning

1. INTRODUCTION

The web has been turned into one of the most important information sources and knowledge bases for scientific, educational and research purposes. Yahoo, DMOZ and some other web directories use human editors to classify web resources (sites or documents), but with the exponential growth of both the amount and the diversity of the web information, low cost and high speed of automated topic-specific web resources discovering, collecting and classifying are highly desirable [12]. In this paper, we use web site mining [9] to seek a list of authoritative web sites that has the same topics with a given seed site set, and then group these sites into different predefined topic categories. Generally, web site mining includes two phases:

- *Searching* phase: Given an initial topic set *T* or a seed set with topics in *T*, seek out a candidate set S^c from an infinite set S^i by some search strategy or algorithm f_1, namely:
  $f_1 : S^i \to S^c$, where $|S^i| >> |S^c|$.
- *Classification* phase: Assign to each object in S^c one or more class labels from a set of predefined topic categories *C*, namely:
  $f_2 : S^c \to S^l$, where S^l is the final labeled set, and $|S^c| = |S^l|$.

The definition gives us insight that the size of the labeled set S^l mainly depends on the searching phase, while the categorization accuracy of S^l mainly depends on the classification phase. In addition, three issues motivate us to design an effective and efficient web site mining algorithm:

1) *The sampling size of web sites*. The efficiency of web site mining crucially depends on the downloading size of web pages, since many classification algorithms must be performed offline, and download of a remote web site is more expensive than in-memory classification operations [9], as shown in Fig. 1. Hence some sampling algorithm must be introduced into the web site mining to download only a small part of a web site yet with the same or higher classification accuracy.

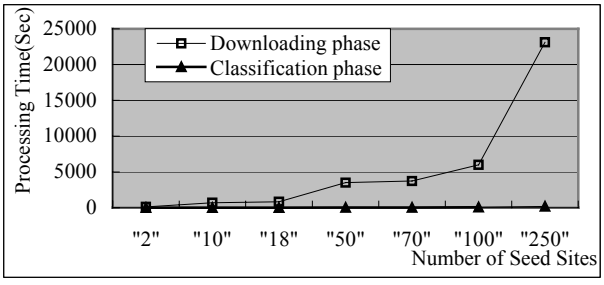


Fig.1 Comparison of processing time of the baseline system between downloading and classification phases with different numbers of seed sites.

2) *The analysis granularity of web site mining*. A majority of existing web site or page categorization algorithms treats a page as an atomic indivisible node

with no internal structure [2]. But as a matter of fact, pages are more complex and may have a number of topics; furthermore, many of them are much *noisy* in the perspective of given topics, such as in banners and advertisements [2]. Hence the web site mining must evolve into a finer level of detail. In other words, pages should be further divided into some *logic snippets* with a single topic. This kind of logic snippets, e.g., DOM (Document Object Model) nodes as proposed by [18, 2], should be treated as the basic analysis units in web site mining. In this paper, we will describe a multiscale representation of web sites, which includes the site, page and DOM node levels (This paper uses *level* and *scale* interchangeably). The multiscale representation not only facilitates denoising the content of web sites at different levels, but also captures the underlying topic dependencies between nodes across levels.

3) *The representation structure of web sites*. Typically, there are three approaches in web mining algorithms: superpage [15, 9], topic vector [9], and tree or graph representation [9, 19]. Among the three kinds of algorithms, the tree approach is more suitable for characterizing the semantic structure of web sites. Therefore, in this paper, we represent a web site as a 3-D multiscale site tree. In this model, all page nodes within the site or all DOM nodes within a page are hierarchically linked by a tree (we refer to them as the page tree and DOM tree respectively), and connecting vertically nodes in the three levels yields a multiscale tree. Meanwhile, four kinds of context models are presented to characterize the intra- and inter-scale topic dependencies between nodes.

On the basis of the above site representation model, this paper proposes a multiscale approach for web site mining. In our approach, we exploit the hidden Markov tree (HMT) model, which was originally introduced to model statistical dependencies between wavelet coefficients in signal processing [6], to capture the intrascale contextual dependencies between nodes in page trees or DOM trees. Therefore, we may apply the HMT-based classification algorithm twice to creating the initial classification results of web sites without regard to the interscale contextual dependencies between nodes across scales. However, in order to overcome the shortcoming of over-localization [13] in the pre-classification of fine-grained DOM nodes, we present an intrascale and interscale fusion algorithm to refine the initial classification results using a coarse-to-fine recursion through scales. Consequently, the multiscale web site classification is computationally efficient by three steps, i.e., raw classification, interscale fusion and re-classification for final results. Moreover, for further improving classification accuracy while reducing the classification overheads, we introduce a two-stage text-based denoising procedure to remove the noisy DOM nodes or pages within web sites and an entropy-base approach to dynamically prune the site trees. The details of the algorithms will be discussed in the following sections.

We evaluate our approach for the web site mining tasks in practice with different numbers of seed sites. And the baseline system is based on the superpage classification approach with a fixed downloading depth of web sites. Experiments show that our approach achieves in average 16% improvement in classification accuracy and 34.5% reduction in processing time over the baseline system.

Compared with the previous web site mining algorithms presented in [15, 9, 19], the main contribution of our approach is to propose the multiscale tree-structured representation model and context-based multiscale classification for web sites. The literature [15] utilized the superpage method, which has been proved to perform poorly. The literature [19] represented a web site as a graph, and employed hyperlink-based classification approach. Experiments show that this approach is more suitable for assistant tasks, such as topic-specific crawling. Although inspired by some concepts introduced in [9], our study has several distinct features. First, we use a multiscale tree-structured representation and multiscale classification framework for web sites. Second, we employ the more comprehensive context models to make use of all correlative semantic clues for site categorization. Third, we employ the relative entropy rather than the variance of the conditional probabilities to construct the dynamic pruning strategy for site trees. In addition, we introduce a comprehensive denoising step to purify the content of web sites in different levels, so as to achieve higher accuracy.

The rest of this paper is organized as follows. In Section 2 we propose the multiscale site tree model. In Section 3 we simply review the HMT model, and then present the HMT-based classification algorithm. In Section 4, we discuss the context-based interscale fusion method, the two-stage denoising procedure and the entropy-base pruning strategy, and then present the framework of our multiscale web site mining algorithm. Experiment design and results will be described in Section 5. Finally, Section 6 concludes this paper.

2. MULTISCALE TREE REPRESENTATION MODEL OF WEB SITES

The representation model of web sites affects the efficiency of the web site mining algorithm. The superpage method just represents a web site as a set of terms or keywords, and directly applies pure text classifiers to web pages and sites [15]. As a result, this method performs poorly and is usually applied to constructing baseline systems. Analogously, the topic vector approach [9] that represents a web site as a topic vector (where each topic is defined as a keyword vector), is essentially a two-phase keyword-based classification. On the other hand, the tree-based representation model [9] can effectually utilize the semantic structure of sites and local contexts, and more importantly, can transform the sampling size issue of web sites to the pruning problem of the site trees. However, the existing tree-based web site

mining methods [9] employed the keyword-based text classifiers for the pre-classification of pages and thus the noise information within pages would still affect the final classification accuracy of web sites. Therefore, this paper extends the existing tree representation model to an extra level of resolution (DOM nodes) and proposes a multiscale tree–structured representation model for web sites. The model is based on the following assumptions:

Assumption 1 (Tree Structure Assumption)*: The structure of most web sites is more hierarchy-like than network-like [9]. Furthermore, each HTML/XML page can be represented as a DOM tree [18].*

Thus, we define the web site as follows:

Definition 1 (Site Structure Model): A web site can be represented as a page tree $T_P = T(P, E)$, where $P = \{p_1, \cdots, p_n\}$, the root p_1 is the starting page of the site, and $\forall p_i \in P$ is a HTML/XML page within the site. Furthermore, p_i can be represented as a DOM tree, i.e., $p_i = DOM_i(DN, DE)$. A link between p_i and p_j is represented by the directed edge $(p_i, p_j) \in E$, where p_i is the parent node of p_j, and p_j is one of the children of p_i. Hence a web site can be further represented as a multiscale tree $T_M = T(\{DOM_i(DN, DE)\}, E)$.

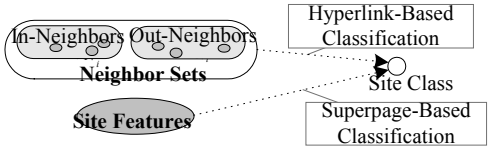
To build the page tree, a *breadth-first search* will be performed. In our application, we only sample the pages located "below" the starting page. For example, if the URL of the starting page is http://phys.cts.nthu.edu.tw/en/index.php, we only sample the pages sharing the base URL http://phys.cts.nthu.edu.tw/en/.

Therefore, there are three kinds of web site structure models to be utilized in web site mining algorithms:

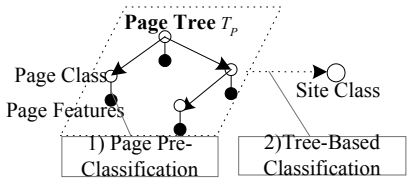
1) *The whole site as the atomic analysis node*. The hyperpage and hyperlink-based classification methods can be used, as shown in Fig 2a.

2) *The whole page as the atomic analysis node and the page tree T_P as the site structure model*. The literature [9] employed this approach, as shown in Fig 2b.

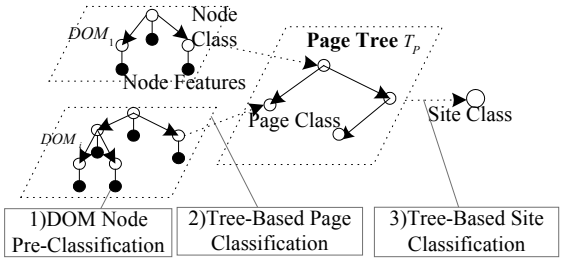
3) *DOM nodes as the atomic analysis nodes and the multiscale tree T_M as the site structure model*. Two-phase tree-based classifiers can be used, as shown in Fig 2c. However, few features extracted from DOM nodes and lack of local context cause poor classification accuracy of DOM nodes. To overcome the over-localization issue [13] in the classification of fine-grained analysis nodes, a multiscale context model and the multiscale classification method should be introduced to web site mining for exploiting both intra- and inter-scale topic dependencies between nodes.



a)The whole site as th atomic analysis node



b)The whole pages as the atomic analysis nodes



c)DOM nodes as the atomic analysis nodes

Fig 2. Three kinds of analysis granularities and corresponding basic classification methods in web site mining, where dashed and solid arrows represent classification operations and topic dependences between nodes respectively. In fact, there are dashed arrows from every lower node (i.e. DOM or page node) to the corresponding parent node (i.e., page or site node), and the tree-based classification operation could be represented by the inference in Bayesian networks.

In hyperlink environment, links contain high-quality semantic clues to a page's topic, and such semantic information can help achieve even better accuracy than that possible with pure keyword-based classification [12]. In this paper we refer to the semantic information surrounding links in a page as the *context* of the page, and generalize this concept to sites and DOM nodes. Therefore, according to Web link topology [1, 19], we have the following assumption:

Assumption 2 (Context Assumption)*: In web mining, context is treated as the nodes topically related to the analysis node. Context information is helpful to improve the classification accuracy of analysis nodes* [3, 4, 13].

According to Markov Random Field (MRF) theory [3, 4] and the above site structure model, we can define the following four kinds of context models used in web site mining:

Definition 2 (Site Context Model *SC*): Each web site is topically related to its *tightly linked* neighbors. Let N_i be the tightly linked neighbors of S_i, then the site context model of S_i is $SC(S_i) = N_i \approx N_i^K = N_i^I \cup N_i^O$, where N_i^K are pre-classified sites in N_i, N_i^I and N_i^O are the in-neighbors and out-neighbors of S_i within N_i^K [3],

$$P(C_{S_i} \mid \{C_{S'} \mid S' \neq S_i\}) = P(C_{S_i} \mid \{C_{S'} \mid S' \in SC(S_i)\}) \quad (1)$$

where C is hidden class variable.

Specially, if S_i is an isolated web site or $N^K = \phi$, then $P(C_{S_i} \mid \{C_{S'} \mid S' \neq S_i\}) = P(C_{S_i})$, i.e., we cannot induce the class information of S_i according to its linked neighbors.

Definition 3 (Page Context Model *PC*): Each web page or hypertext is topically related to its in-linked and out-linked pages [1, 3]. In the page tree T_P, p_i's parent node

ρ_i is one of its in-link page, p_i's children ($p_{c_1},\cdots,p_{c_{n_i}}$) are its out-link pages, then the page context model of p_i is $PC(p_i)=\{\rho_i,p_{c_1},\cdots,p_{c_{n_i}}\}$, where

$$P(C_{p_i}\mid\{C_{p'}\mid p'\neq p_i\})=P\{C_{p_i}\mid C_{\rho_i},C_{p_{c_1}},\cdots,C_{p_{c_{n_i}}}\} \quad (2)$$

Definition 4 (DOM Context Model *DC*): Each DOM node is topically related to its parent and children nodes in the DOM tree [2]. Namely, the DOM context model of DN_i is $DC(DN_i)=\{\rho(DN_i),DN_{c_1},\cdots,DN_{c_{n_i}}\}$, where

$$P(C_{DN_i}\mid\{C_{DN'}\mid DN'\neq DN_i\})=P\{C_{DN_i}\mid C_{\rho(DN_i)},C_{DN_{c_1}},\cdots,C_{DN_{c_{n_i}}}\} \quad (3)$$

Definition 5 (Mutliscale Context Model *MC*): The topic dependencies between the parent and the child scales can be used to refine raw classification results of nodes at the child scale [13, 10, 4]. For example, the class information of pages can provide prior information to its children DOM nodes since children nodes are likely to be in the same class as their parent. In the multiscale site tree T_M, let W_i^l be the i^{th} node at the l^{th} level (l=1, 2, 3 correspond to the site, page and DOM node levels, respectively), ρ_i^{l-1} be its parent node at the (l - 1)$^{\text{th}}$ level, then the multiscale context model of W_i^l is $MC(W_i^l)=\{\rho_i^{l-1}\}$, where

$$P(C_{W_i^l}\mid\{C_{W^{l'}}\mid l'<l\})=P\{C_{W_i^l}\mid C_{\rho_i^{l-1}}\} \quad (4)$$

Without loss of generality, if we view the neighbors N_i of the site S_i as the 0$^{\text{th}}$ level nodes, then the site context model can be considered as a particular case of the multiscale context model. Therefore, we refer to the site and multiscale context models as the interscale context models, and correspondingly, the page and DOM context models as the intrascale context models.

The above site structure model and context models constitute our web site representation model. Fig 3 shows its graph model. Depicted as a 3-D multiscale tree, the model represents the structure of web sites as $T_M=T(\{DOM_i(DN,DE)\},E)$, and captures the topic dependencies (NOT link structure) between nodes at the same level according to the intrascale context models. Moreover, a multiscale tree connects vertically nodes in the three levels, and their topic dependencies are modeled as the interscale context models.

The most important advantage of this representation is to make multiscale classification architecture possible for web site mining. In general, if a web site is of a pure class, the classification at coarser scales is more reliable due to their richness in feature information. However, since a web site often contains multiple topics and is much noisy, classification performed at large scales, such as at the site level or the page level, is crude. On the other hand, fewer features extracted from DOM nodes and lack of local context would also cause poor classification results of the fine-grained nodes. The multiscale classifier thus exploits the dependencies between the parent and the child scales to refine the raw classification results of the fine-grained nodes, so as to get both reliable and accurate classification.

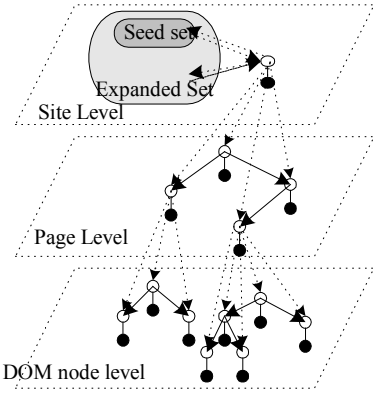


Fig 3. The multiscale tree-structured representation model of web sites. The dashed and solid arrows represent inter- and intra-scale topic dependences between nodes respectively.

In the following sections, we present a multiscale web site mining algorithm based on the above site representation model.

3. THE HMT-BASED CLASSIFICATION ALGORITHM

In the above site representation model, tree is the basic data structure. A web site is represented as a page tree and a page is also represented as a DOM tree. Both in the page context model and in the DOM context model, the parent and the children nodes together constitute the intra-scale context of the analysis node. Hence, we can not choose the 1-order Markov Tree in [9], which only models the context of the analysis node as its parent node, but the hidden Markov tree model proposed in signal processing as the statistical model of page trees and DOM trees.

3.1 The HMT Model

The HMT was initially introduced to model the statistical dependencies between wavelet coefficients in signal processing [6, 16, 17]. Meanwhile, M. Diligenti et al also proposed the hidden tree Markov model (HTMM) for learning probability distributions defined on spaces of labeled trees [7, 8]. In the essence of statistics, the two models have no differences. Hence this paper uses HMT as their general designation and applies it on the web site classification.

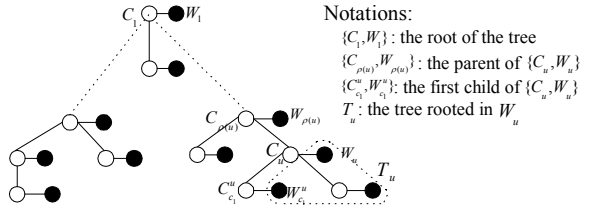


Fig 4. Graph Representation of the hidden Markov tree. Black and white nodes represent observation and hidden state variables, respectively.

Let $W=(W_1,\cdots,W_n)$ refer to the observed data, which is

indexed by a tree rooted in W_1 (See Fig. 4). Let $C=(C_1,\cdots,C_n)$ be the hidden class labels that have the same indexation structure with W. Then an HMT model λ [6] is specified via the distribution $\pi=(\pi_k)_{k\in\{1,\cdots,K\}}$ for the foot node C_1, the state transition matrices $A=(a^{rm}_{\rho(u),u})$ where $a^{rm}_{\rho(u),u}=P(C_u=m\,|\,C_{\rho(u)}=r)$ and the observation probabilities $B=\{b_j(k)\}$. The HMT model has the following two properties [6]:

- *The conditional independence property*. Each observation W_i is conditionally independent of all other random variables given its state C_i, i.e.,

$$P(W_1,\cdots,W_n\,|\,C_1,\cdots,C_n)=\prod_{u=1}^{n}P(W_u\,|\,C_1,\cdots,C_n)$$
$$=\prod_{u=1}^{n}P(W_u\,|\,C_u) \qquad (5)$$

- *Markov tree property*. Given the parent state $C_{\rho(u)}$, the nodes $\{C_u,W_u\}$ are independent of all other nodes except for C_u's descendants, i.e.,

$$P(C_u\,|\,C_{u'}\neq C_u)=P(C_u\,|\,C_{\rho(u)},C^u_{c_1},\cdots,C^u_{c_{n_u}}) \qquad (6)$$

By comparing the formula (6) with (2) and (3), we conclude that the HMT model can exactly characterize the intrascale contextual dependencies between nodes within the page trees and DOM trees.

Similar to HMMs, there are three problems associated with HMTs, i.e., likelihood determination, state estimation and training problems. The likelihood can be calculated through the *upward-downward* procedure [6], and the second problem can be efficiently accomplished by Viterbi algorithm. For the third problem we can resort to the iterative Expectation Maximization (EM) algorithm. In our application, however, the incremental learning is very important because the disequilibrium distribution among different classes of samples (See Fig. 5) makes us exploit incrementally available data in the web site mining tasks as new samples to re-train the models. Hence in this paper we adapted the incremental EM algorithm for HMMs presented in [11] to train the HMT models.

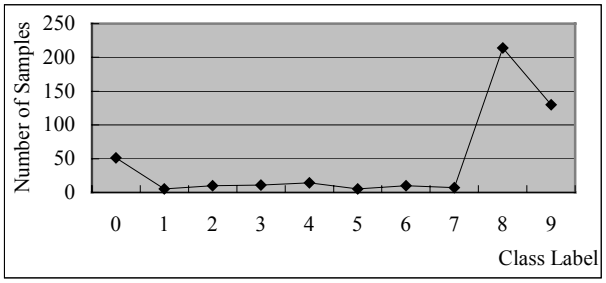


Fig 5. The disequilibrium distribution of training samples in our experiments.

To verify the incremental EM HMT training algorithm, we performed a simple experiment using discrete observation HMT with random parameterization and sampling. The training samples were divided into 3 and 10 subsets. Fig. 6 shows the comparison of log-likelihood functions for the incremental and batch EM HMT training algorithms. Results indicate that the incremental approach has numerically stable iterative scheme and even converges faster than the batch version.

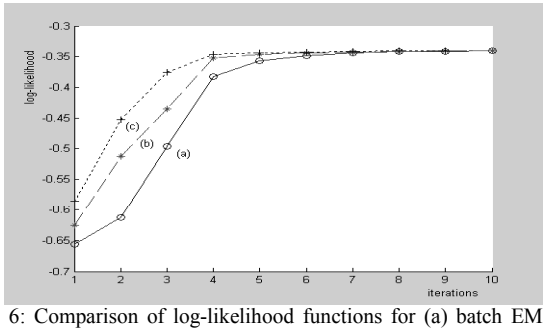


Fig 6: Comparison of log-likelihood functions for (a) batch EM HMT training and incremental EM HMT training with (b) 3 and (c) 10 subsets.

3.2 HMT-Based Classifier

Once the HMT model is trained, we may utilize Maximum A Posterior (MAP) principle to construct the HMT-based classifier.

Let W be all observed data of a page tree or a DOM tree, T_1. The classifier can be expressed as:

$$C_i=\arg\max_i P(C_i\,|\,W,\lambda) \qquad (7)$$

Using the downward-upward algorithm [6], we can get

$$P(C_i=m\,|\,W,\lambda)=\frac{P(C_i=m,W\,|\,\lambda)}{P(W\,|\,\lambda)}=\frac{\beta_i(m)\alpha_i(m)}{\sum_{k=1}^{K}\beta_i(k)\alpha_i(k)} \qquad (8)$$

where $\beta_i(k)=P(T_i\,|\,C_i=k)$ is the upward variable, and $\alpha_i(k)=P(C_i=k,T_{1\setminus i})$ is the downward variable.

To simplify the parameter estimation, we model the hidden state variables as class labels of analysis nodes and then train two HMT-based classifiers for page trees and DOM trees respectively. Another optional method is to train an HMT model for each class of page trees and DOM trees and then use Maximum Likelihood (ML) principle to construct HMT-based classifiers. Here we don't discuss it further.

The HMT-based classifiers can be directly applied to web site classification. When the whole page is treated as the atomic analysis node, the classification procedure is shown in Fig 2b; and when the DOM node is treated as the atomic analysis node, the two-phase classification procedure is shown in Fig 2c.

4. THE MULTISCALE WEB SITE MINING ALGORITHM

As discussed before, the poor raw classification results of DOM nodes might decrease the accuracy of two-phase HMT-based classification. Hence in this section we will investigate how to combine the interscale context models with the HMT-based classifiers to improve the raw classification accuracy. After that, another two issues, i.e.,

denoising and pruning, will be discussed. At last of this section, we will present the framework of the multiscale web site mining algorithm.

4.1 Context-Based Interscale Fusion

In the two-phase HMT-based classification, only intra-scale context models are utilized. In the following, the interscale context models will be used to refine the results of raw classification.

Let W_i^l be the i^{th} node at the l^{th} level (l = 1, 2, 3), $MC_i^1 = SC(W_i^1)$ (l = 1) or $MC_i^l = MC(W_i^l)$ (l = 2, 3) be its interscale context. Let $P(C_i^l \mid W_i^l)$ be the raw classification result of W_i^l computed by the two-phase HMT-based classifier (when l=3, it is the pre-classification result of the DOM node computed by the keyword-based text classifier), then

$$P(C_i^l \mid W_i^l, MC_i^l) = \frac{P(W_i^l \mid C_i^l, MC_i^l) P(C_i^l \mid MC_i^l)}{P(W_i^l \mid MC_i^l)}$$

(Since W_i^l is independent of MC_i^l given C_i^l)

$$= \frac{P(W_i^l \mid C_i^l) P(C_i^l \mid MC_i^l)}{P(W_i^l \mid MC_i^l)}$$

$$= \frac{P(W_i^l)}{P(W_i^l \mid MC_i^l)} \cdot \frac{P(C_i^l \mid W_i^l) P(C_i^l \mid MC_i^l)}{P(C_i^l)}$$

(Here $\frac{P(W_i^l)}{P(W_i^l \mid MC_i^l)}$ can be viewed as a constant α [20])

$$= \frac{\alpha P(C_i^l \mid W_i^l) P(C_i^l \mid MC_i^l)}{P(C_i^l)} \quad (9)$$

where $P(C_i^l)$ is the prior probability of class C_i^l, $P(C_i^l \mid MC_i^l)$ is the contextual probability. To simplify the parameter estimation, the values of $P(C_i^l)$ and $P(C_i^l \mid MC_i^l)$ are obtained by averaging over all the nodes at that scale. Therefore, if $|C|$ denotes the number of topic classes, then we need to estimate totally $2 \times 3 \times |C|$ parameters. We can easily estimate $P(C_i^l)$ as the relative frequency of l^{th} level nodes in the class C_i^l. Hence, the key problem is to estimate $P(C_i^l \mid MC_i^l)$.

In image segmentation, there are two methods for calculating the contextual probabilities. The Contextual Labeling Tree (CLT) is the common way [5, 10]. A CLT is a tree-structured graph in which a context node is augmented to each observation node as a function of the other nodes. By CLT model, more than one context models can be combined sequentially to obtain better classification results [10]. Another method is to use Class Probability Tree (CPT) [4], which represents a sequence of decisions or tests that must be made in order to compute the contextual probabilities. In this paper, we calculate $P(C_i^l \mid MC_i^l)$ in the following two cases:

1) The contextual probability $P(C_i^1 \mid MC_i^1)$ at the site level: According to the site context model $MC_i^1 = SC(S_i) = N_i \approx N_i^K = N_i^I \cup N_i^O$, we get:

$$P(C_i^1 \mid MC_i^1) = P(C_i^1 \mid N_i^K) = \frac{P(N_i^K \mid C_i^1) P(C_i^1)}{P(N_i^K)} \quad (10)$$

where $P(N_i^K)$ is the relative frequency of pre-classified neighbors N_i^K in N_i, and $P(N_i^K \mid C_i^1)$ is computed by the following formula [3]:

$$P(N_i^K \mid C_i^1) = \prod_{\delta_j \in N_i^I} P(C_j \mid C_i^1, j \to i) \prod_{\delta_k \in N_i^O} P(C_k \mid C_i^1, i \to k) \quad (11)$$

2) The contextual probability $P(C_i^l \mid MC_i^l)$ (l=2, 3) at the page and DOM node levels: According to the multiscale context model, $MC_i^l = MC(W_i^l) = \{\rho_i^{l-1}\}$. $P(C_i^l \mid MC_i^l)$ can be computed by the CLT as presented in [5, 10]. However, to reduce the complexity of computation, this paper approximates it by $P(C_i^{l-1} \mid \rho_i^{l-1})$, i.e.,

$$P(C_i^l \mid MC_i^l) = P(C_i^{l-1} \mid \rho_i^{l-1}) \ (l=2, 3) \quad (12)$$

Now, the new classifier is defined as follows:

$$\hat{C}_i^l = \arg\max_i P(C_i^l \mid W_i^l, MC_i^l) \quad (13)$$

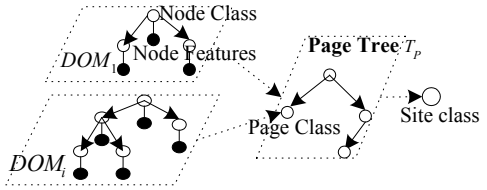
Therefore, the multiscale classification procedure in the web site mining algorithm includes the following four steps, as shown in Fig 7:

Step 1 (Pre-Classification): Classify all DOM nodes using keyword-based text classifiers;

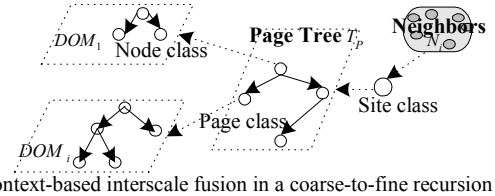
Step 2 (Raw Classification): Classify pages and sites using HMT-based classifiers in a fine-to-coarse recursion through scales, as shown in Fig 7a;

Step 3 (Interscale Fusion): Refine the previous results using context-based interscale fusion algorithm in a coarse-to-fine recursion through scales, as shown in Fig 7b;

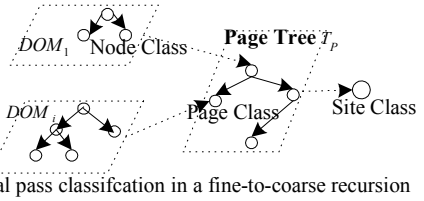
Step 4 (Final Pass Classification): Re-classify pages and sites using HMT-based classifiers in a fine-to-coarse recursion through scales, and get the final classification results, as shown in Fig 7c.



a) HMT-based classification using intrascale context models in a fine-to-coarse recursion



b) Context-based interscale fusion in a coarse-to-fine recursion



c) Final pass classifcation in a fine-to-coarse recursion

Fig 7. The multiscale web site classification procedure

4.2 Denoising and Pruning

As discussed before, to obtain high classification accuracy in the web site mining, two tasks should be performed, i.e., denoising and pruning.

The task of denoising is to remove the DOM nodes or pages that are irrelevant to the query topics or cannot be identified by text-based classifiers, including animated introductions and frames, banners, navigation panels, and advertisements, etc. [2]. Hence, it is natural to utilize text-based denoising method. In this application, we use the two-stage procedure to purify the content of web sites, namely, text-based denoising method is performed at the DOM node level at first, and if a majority of DOM nodes within a page is marked for removing, then the page is removed in whole. Considering the high dimensionality of centroid vector and the limited statistical information within a DOM node cause most similarity scores are near to zero, we exploit the thesaurus-based rather than the centroid vector based text-denoising method. As a classical classification approach widely used in information analysis field, the thesaurus-based method determines the pertinence of a DOM node to a given topic by analyzing the occurrence frequency of the topic-specific keywords and terms in that node. In our experiments we employed the *Physics Subject Thesaurus* with 9181 terms as the thesaurus for text-based denoising. Furthermore, 54311 keywords were extracted from a large amount of textual data of *Physics Digest* to enrich the thesaurus. Experiments showed in our application settings the thesaurus-based method outperformed the centroid vector approach.

On the other hand, the pruning process for reducing the sampling size of web sites is more complex. The literature [9] exploited the variance of the conditional probabilities over the set of all web site classes to measure the importance of a path for site classification and then proposed a pruning algorithm based the variance and the path length. To capture data structure beyond second order (variance) statistics, in this paper, we employ the *relative entropy* or *Kullback Leibler distance* [14] to model the 'distance' between the distributions embodied by the original model and by the pruned model. In addition to the site tree structure assumption, our pruning approach is based on the following assumptions:

Assumption 3 (Assumption on sampling necessity)*: In web mining, download of a remote web page is more expensive than in-memory operations [9].*

This assumption has been verified by the experiment shown in Fig 1. It not only shows the sampling importance in web mining, but also enlightens us that we might reduce dramatically the downloading time and increase somewhat local in-memory operations so as to optimize the total processing time.

Assumption 4 (Assumption on sample size)*: Web site classification needs to download web pages within the site. However, after the downloaded pages are more than a fixed quantity, to download more pages cannot improve the classification accuracy.*

It is not clear how many pages are *sufficient* for web site classification in order to keep a comparatively high accuracy. Intuitively, there are cases where a whole subtree and the path leading to it do not show any clear class membership at all [9], hence features extracted from these pages cannot be helpful to improve classification accuracy. This paper uses Kullback Leibler distance [14] to measure whether adding a page to the page tree of a web site will result in a reduction in the uncertainty of classification results or not. Therefore, we propose the following dynamic pruning strategy for web site trees (See Fig 8):

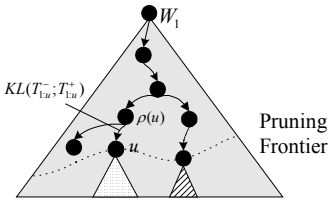


Fig 8. Pruning the page tree of a web site. All pages below the pruning frontier will not be downloaded.

Let u be the current downloading page, $T_{1:u}^-$ and $T_{1:u}^+$ denote the page trees of a web site before and after downloading u, and $P(C_i \mid T_{1:u}^-, \lambda_P)$ and $P(C_i \mid T_{1:u}^+, \lambda_P)$ be the likelihood functions corresponding to $T_{1:u}^-$ and $T_{1:u}^+$ given the HMT model λ_P at the page level. The KL distance from $T_{1:u}^-$ to $T_{1:u}^+$ is given by

$$KL(T_{1:u}^-, T_{1:u}^+) = \sum_i P(C_i \mid T_{1:u}^-, \lambda_P) \log \frac{P(C_i \mid T_{1:u}^-, \lambda_P)}{P(C_i \mid T_{1:u}^+, \lambda_P)} \qquad (14)$$

Then the dynamic pruning strategy can be defined:

If $(KL(T_{1:u}^-, T_{1:u}^+) \geq depth(T_{1:u}^+) \cdot \Delta)$ *then add* u *to the page tree, else suppress the growth of the tree to node* u.

Where $depth(T_{1:u}^+)$ is the depth of the page tree $T_{1:u}^+$, and Δ is the convergence parameter used in the HMT training at the page level. Similar to the pruning strategy presented in [9], this pruning approach becomes less sensitive with increasing $depth(T_{1:u}^+)$. Our experiment will show the entropy-based approach can improve classification accuracy by downloading only a small part of a web site.

4.3 Algorithm Framework

Now, the framework of the multiscale web site mining algorithm is as follows:

Algorithm 1: The multiscale web site mining algorithm

Input: a labeled seed site set $ST^{(0)} = \{S_i\}$, a graph G defined by the

sites in $ST^{(0)}$ and links, a set of topic classes $C = \{C_i\}_{i=1,\cdots,|C|}$ and a process termination condition Γ.

Initialization:

Let the initial expended site set $ST^{(i)} = \phi$, the training set $T = ST^{(0)}$, the counter *r*=1.

Repeat

1. Training Phase: Given the training set T,

1.1 Train the parameters for the HMT-based page tree classifier, $\lambda_P^{(r)}$ by the incremental EM;

1.2 Train the parameters for the HMT-based DOM tree classifier, $\lambda_D^{(r)}$, by the incremental EM;

1.3 Calculate the parameters for interscale fusion algorithm.

2.Seaching Phase: A hyperlink-based focused crawler is utilized to discover the candidate site set CT in which sites share the similar topics with the sites in the seed set $ST^{(0)}$. The hyperlink graph $G' = \{(v_i, v_j) \mid v_i, v_j \in ((\bigcup_{i=0,r} ST^{(i)}) \bigcup CT)\}$ is recorded.

3.Downloading Phase:

For $\forall S_i \in CT$ **do**

3.1 Initialize *p* with the starting page of S_i.

While the downloading process does not terminate **do**

3.2 Download the page *p*;

3.3 Build the DOM tree for *p*, and perform the *text-based denoising* for the DOM tree;

3.4 Build the page tree $T_{1:p}$ for S_i using currently downloaded pages;

3.5 Raw Classification (Fine-to-Coarse):

3.5.1 Classify all DOM nodes of *p* with the *Keyword-based classifier*;

3.5.2 Classify *p* with the *HMT-based DOM tree classifier*;

3.5.3 Classify S_i with the *HMT-based page tree classifier* on $T_{1:p}$.

3.6 Pruning: Apply *the entropy-based dynamic pruning algorithm* to deciding whether to extend *p* to its subtrees or not.

3.7 Set *p* be the next page according to the *breadth-first search strategy*.

End While.

End For.

4. Multiscale Classification Phase:

For $\forall S_i \in CT$ **do**

4.1 Interscale Fusion (Coarse-to-Fine):

For *l*=1 to 3 **do**

4.1.1 Calculate MC_i^l for each node;

4.1.2 Calculate $P(C_i^l \mid W_i^l, MC_i^l)$;

End For

4.2 Final Pass Classification (Fine-to-Coarse):

4.2.1 Re-Classify all *p* in S_i with the *HMT-based DOM tree classifier*;

4.2.2 Re-Classify S_i with the *HMT-based page tree classifier* on T_i (T_i is the pruned site tree of S_i).

4.3 Update the expended site set: $ST^{(r)} \leftarrow ST^{(r)} \bigcup \{S_i\}$.

End For.

5. $T = ST^{(r)}, r \leftarrow r+1$.

Until Γ.■

5. EXPERIMENTS AND RESULTS

The goal of our work is to develop an intelligent information analysis tool for topic-specific web resources, iExpert, in Chinese Science Digital Library Project. We evaluated our approach for web site mining tasks in practice with different numbers of seed sites. Total 528 practical physics web sites were used as seed sites. They had been downloaded to local server completely, and labeled by domain experts according to the *Physics Subject Classification*, which composes of 10 classes and 71 subclasses. It should be noted that the minor distinguishability between some classes in the class hierarchy increased the difficulty of classification tasks (See Table 1). The baseline system was based on the bilingual kernel-weighted KNN classifier, using the superpage classification approach with a fixed downloading depth of web sites. A hyperlink analysis program and a web focused-crawler were used for both the baseline system and the multiscale web site mining algorithm (MSM for short). All experiments were run in the following environments: 800MHz CPU, 256MB RAM and shared 2M LAN bandwidth.

Table 1. The First layer classes in *Physics Subject Classification*

No	Class Name
0	General
1	The physics of elementary particles and fields
2	Nuclear physics
3	Atomic and molecular physics
4	Classical areas of phenomenology
5	Fluids, plasmas and electric discharges
6	Condensed matter: structure, thermal and mechanical properties
7	Condensed matter: electronic structure, electrical, magnetic, and optical properties
8	Cross disciplinary physics and related areas of science and technology
9	Geophysics, astronomy and astrophysics

The evaluation metrics for the web site mining tasks are *spotting capability*, *classification accuracy* and *processing time*. The spotting capability is defined as the ratio between the number of new web sites, N_{new}, and seed sites, N_{seed}, i.e.:

$$spotting = \frac{N_{new}}{N_{seed}} \times 100\% \quad (15)$$

Throughout all experiments, the numbers of seed sites were 2, 10, 18, 50, 70, 100, 250, respectively (These seed sites were also used as training samples). And the average spotting capability was 187%. Meanwhile, we use the classification accuracy as the main measure to compare the proposed method with other previous work. The classification accuracy is defined as:

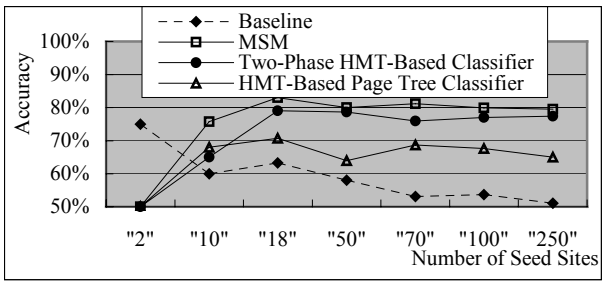
$$accuracy = \frac{1}{10} \sum_{i=0}^{9} \frac{M^{(i)}}{N_{new}^{(i)}} \times 100\% \quad (16)$$

where $N_{new}^{(i)}$ is the number of the new spotting sites with the class *i* (*i*=0~9) and $M^{(i)}$ is the number of the accurately classified sites with the same class attribute.

Fig. 9 shows the comparison of classification accuracy of different classifiers given different numbers of seed sites. Besides the baseline system and the MSM algorithm, we also employed the HMT-based page tree classifier (as depicted by Fig 2b), the two-phase HMT-based classifier (as depicted by Fig 2c) and the 0-order Markov tree classifier (as presented in [9]) to classify the new

discovering web sites. Not surprisingly, although performed poorly in small size of training set, the MSM algorithm had the best accuracy, with nearly 16.7% improvement over the baseline superpage approach. We also noticed that since the MSM algorithm and the two-phase HMT-based classifier carried out denoising operations at both the DOM node and the page levels, they clearly outperformed than the HMT-based page tree classifier that only denoised at the page level by about 11% and 7% respectively. This conclusion confirmed that the noise information in web pages and web sites was one of the main factors to cause the comparatively low classification accuracy. Meanwhile, the MSM algorithm outperformed the two-phase classifier by about 3.9%, mainly because the former refined the classification results of the latter using the interscale context models.

On the other hand, the 0-order Markov tree classifier provided only an accuracy of about 69.3%, which is 6.4% and 2.6% less than the MSM algorithm and the two-phase HMT-based classifier respectively, but 4.4% more than the HMT-based page tree classifier. Compared with the two algorithms that were based on the multiscale site representation, the Markov tree classifier utilized the uni-scale tree representation model, simpler context models and denoising strategy at only page level, thus had the worse accuracy. However, the Markov tree classifier outperformed than the HMT-based page tree classifier due to fewer parameters though they utilized the same site structure model. We also noticed that in our experiments the accuracy of the Markov tree classifier was much less than 87% described in [9]. A possible reason is that in our class hierarchy the minor difference between some classes causes the classification errors easily.



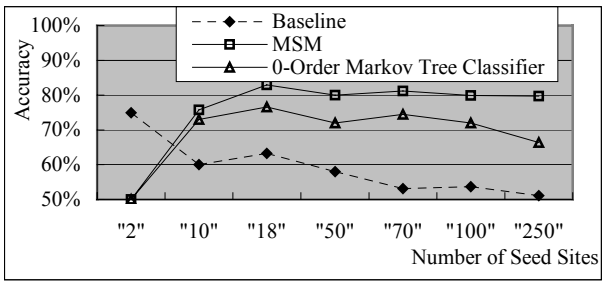


Fig.9 Comparison of the accuracy with different numbers of seed sites: (a) Comparison of the baseline superpage classifier, the HMT-based page tree classifier, the two-phase HMT-based classifier and the MSM algorithm; (b) Comparison of the baseline, the MSM algorithm and the 0-Markov tree classifier proposed by [9].

We have also compared the processing time of the baseline system and the MSM-based system, and found that on the average, the MSM-based system had saved 46.7% downloading time (See Fig. 10) but spent more 66.2% classification time compared with the baseline system, which always downloaded a fixed three levels of pages from web sites. Totally, the MSM-based system saved 34.5% processing time in the whole web site mining procedure. This result confirmed the assumption 3, i.e., we can obtain dramatic reduction of total processing time at the cost of increasing somewhat local in-memory operations. Furthermore, comparison of the classification accuracy of the MSM algorithms with the pruning step and with the fixed download depth shows that pruning step would yield about 5% accuracy improvement with limited increase in the downloaded data (See Fig 11). This is because the pruning strategy purposefully imposed on somewhat controls on the sampling process. These results show that the entropy-based pruning algorithm is efficient.

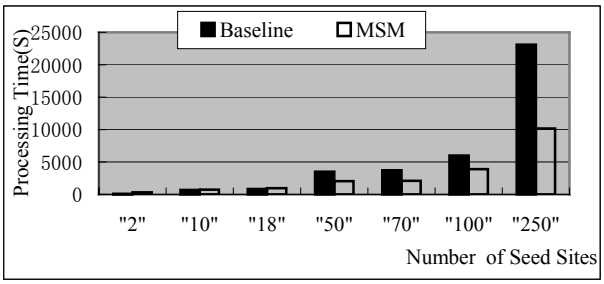


Fig.10 Comparison of the processing time of the baseline and the MSM- based systems on downloading phase.

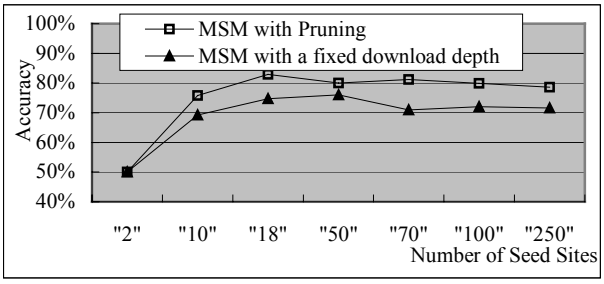


Fig.11 Comparison of the classification accuracy of the MSM algorithms with and without using entropy-based pruning method.

To sum up, the multiscale web site mining algorithm can offer high classification accuracy and efficient processing time. Nevertheless, the performance should be further improved when limited training samples are available.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we discussed three issues to be solved for designing the effective and efficient web site mining algorithm, i.e., the sampling size, the analysis granularity and the representation structure of web sites. By extending the existing representation and classification methods of web sites to an extra level of resolution (DOM nodes), we proposed a multiscale tree-structured representation model for web sites and presented a novel multiscale web site mining approach, which contains an HMT-based classification algorithm, a context-based interscale fusion algorithm, a two-stage text-based denoising procedure and an entropy-base pruning strategy.

Experiments showed that our approach obtained some improvements over the baseline as well as other existing algorithms.

Some concepts and methods originally proposed in signal processing were extended into web site mining in this paper, such as multiscale data representation and classification, denoising and context models, etc. The encouraging results motivate us to further investigate more effective representation models and mining algorithms incorporating the textual and multimedia features to more efficiently discover knowledge from the World Wide Web.

7. ACKNOWLEDGEMENT

This work was supported by "Knowledge Innovation Initiative" of Chinese Academy of Sciences under Grant No. Kgcxz-103. We would like to thank Dr. SiMin He for his help and useful comments.

REFERENCES

[1] Attardi, G., Gulli, A., and Sebastiani, F.: Automatic Web Page Categorization by Link and Context Analysis. In Chris Hutchison and Gaetano Lanzarone (eds.). Proc. of THAI'99, European Symposium on Telematics, Hypermedia and Artificial Intelligence (1999) 105-119.

[2] Chakrabarti, S., Joshi M., and Tawde V.: Enhanced Topic Distillation using Text, Markup Tags, and Hyperlinks. In Proc. of the ACM SIGIR'01, New Orleans, Louisana, USA (2001)

[3] Chakrabarti, S., Dom, B., and Indyk P., Enhanced Hypertext Categorization Using hyperlinks. In Proc. of SIGMOD'98 Seattle, Washington (1998)

[4] Cheng H. and Bouman C. A.: Multiscale Bayesian Segmentation Using a Trainable Context Model. IEEE Trans. on Image Processing, Vol.10, No. 4 (April 2001) 511-525.

[5] Choi, H. and Baraniuk, R. G.: Multiscale Texture Segmentation Using Wavelet-Domain Hidden Markov Models. In Proc. 32nd Asilomar Conference (Nov. 1998).

[6] Crouse, M.S., Nowak, R.D., and Baraniuk, R.G.: Wavelet-Based Statistical Signal Processing using Hidden Markov Models. IEEE Transactions on Signal Processing, Vol. 46 (1998) 886-902

[7] Diligenti, M., Frasconi, P., and Gori, M.: Image Document Categorization Using Hidden Tree Markov Models and Structured Representation. In Singh, S, Murshed, N. & Kropatsch, W. (Eds.) Advances in Pattern recognition - ICAPR 2001. Lecture Notes in Computer Science (2001).

[8] Diligenti, M., Gori, M., Maggini, M., and Scarselli, F.: Classification of HTML documents by Hidden Tree-Markov Models, In Proceedings of the International Conference on Document Analisys and Recognition (ICDAR), Seattle, WA USA (2001) 849—853.

[9] Ester, M., Kriegel, H. P., and Schubert, M.: Web Site Mining: A new way to spot Competitors, Customers and Suppliers in the World Wide Web. In Proc. of SIGKDD02 Edmonton, Alberta, Canada (2002)

[10] Fan G. and Xia, X.-G.: Multiscale Texture Segmentation Using Hybrid Contextual Labeling Tree, in Proc. of the IEEE International Conference on Image Processing (ICIP2000), Vancouver, Canada (Sept. 2000).

[11] Gotoh, Y. Hochberg, M. and Silverman, H.: Efficient Training Algorithms for HMMs Using Incremental Estimation". IEEE Transactions on Speech and Audio Processing, Vol.6, No.6 (1998) 539-548.

[12] Han, J.W. and Chang, K. C.: Data Mining for Web Intelligence. IEEE Computer, Vol. 35, No.11 (2002) 64-70.

[13] Li, J., Gray, R. M.: Context-based Multiscale Classification of Document Images Using Wavelet Coefficient Distributions, IEEE Trans. on Image Processing, Vol. 9, No. 9 (Sep 2000) 1604-1616.

[14] Minh N. D.: Fast Approximation of Kullback-Leibler Distance for Dependence Trees and Hidden Markov Model. IEEE Signal Processing Letters (Apr. 2003).

[15] Pierre, J. M.: On the Automated Classification of Web Sites. Computer and Information Sciences, Vol. 6 (2001)

[16] Romberg, J., Choi, H., Baraniuk, R., and Kingsbury, N., Hidden Markov Tree Models for Complex Wavelet Transforms, IEEE Transactions on Signal Processing (May 2002).

[17] Romberg, J. K., Choi, H., and Baraniuk, R. G.: Bayesian Tree-Structured Image Modeling Using Wavelet-Domain Hidden Markov Models. IEEE Trans. On Image Processing, Vol. 10, No. 7 (Jul 2001).

[18] Stenback, J., Hégaret P. L. and Hors A. L.: Document Object Model (DOM) Level 2 HTML Specification (Version 1.0), W3C Tech Report, http://www.w3.org/TR/2003/REC-DOM- Level-2-HTML-20030109 (2003)

[19] Terveen, L., Hill, W., and Amento, B.: Constructing, Organizing, and Visualizing Collections of Topically Related Web Resources. ACM Trans. On Computer-Human Interaction, Vol. 6 (1999) 67-94

[20] Ye, Z. and Lu, C-C: Unsupervised Multiscale Classification Using Wavelet-Domain Hidden Markov Tree Model in the proceedings of the Student Forum of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'2002), Orlando, Florida, USA (May, 2002).