Framework for Building a High-Quality Web Page Collection Considering Page Group Structure

Yuxin Wang and Keizo Oyama
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430 Japan
{mini_wang,oyama}@nii.ac.jp

Abstract. We proposed a framework for building a high-quality web page collection considering page group structure with two step processes: the rough filtering and the accurate classification. In both processes, we apply the idea of local page group structure that is represented by the relation between a target page and a surrounding page based on the connection types and the relative URL hierarchy. In this paper, we use researchers' homepages as an example of target categories.

In the rough filtering, we proposed a method for comprehensively gathering all potential researchers' homepages from the web with as few noise pages as possible by using property-based keyword lists according to four page group models (PGMs) based on the page group structure. The experiment results show that it reduces the increase of gathered page amount to an allowable level and gathers a significant number of positive pages that could not be gathered with a single-page-based method.

In the accurate classification, we proposed a textual feature set for support vector machine (SVM). The surrounding pages are grouped based on the page group structure, an independent feature subset is generated from each group, and then the feature set is composed by concatenating the feature subsets. An evident improvement of classification performance is shown by an experiment. Using in combination a recall-assured classifier and a precision-assured classifier each of which is obtained by tuning the SVM with the proposed feature set, we next build a three-way classifier to accurately select the pages that need manual assessment to assure the required quality. The effectiveness is shown with the reduction of the manual assessment page number.

Keywords. web page collection, page group model, three-way classifier, quality assurance, precision and recall.

1 Introduction

High quality scholarly information services were maintained with much human work formerly, but with much less today, thanks to the electronic publishing technology. On the other hand, the web is becoming more and more important as a potential information source to add values to such services. Then, what is required first is a web page collection with a guaranteed high quality (i.e., recall and precision); however, it demands a large amount of human work to build because of diversity in style, granularity and structure of web pages, vastness of the web data and sparseness of relevant pages.

Many researchers have investigated search and classification of web pages, etc.; however, most of them are of best-effort type and pay no attention to quality assurance. Thus, we investigate a method to comprehensively build a homepage collection efficiently with assuring both given high recall and high precision. As an example, we mainly focus on researchers' homepages in this paper.

Some research works show that it is generally effective to collect homepages by using the features exploiting link structure, directory structure, document tag structure, and document semantic structure, among others. Taking into account that a homepage are often represented by a logical page group, the information of the surrounding pages in page group structure (local link structure) must be considered in addition to the contents in the entry pages. Therefore, we propose a method to utilize the features considering page group structures for building a high-quality homepage collection with support vector machine (SVM).

Since the web page amount is very large, one problem arising from the method is the high computational cost of its feature extraction. Therefore, we split the process into two steps: rough filtering at first for efficiently narrowing down the candidate page amount with a very high recall, and then accurate classification for accurately classifying the candidate target pages output from the rough filtering with both high recall and high precision. Both processes are realized taking into consideration page group structures and utilizing the useful information based on the web structures so that we can achieve a high classification performance, especially in terms of recall, for meeting the quality requirement for the collection.

2 Related Works

The method proposed in this paper belongs to a web page classification domain, and is closely related to web page search and clustering domains. In these domains, what information sources to use is the first factor and how to use them is the second.

The prior works tried to exploit, besides textual contents, various web-related information sources [2], such as html tags [3, 4, 5], URLs [6, 7, 8], subgraphs of web pages [9, 10], directory structure [10, 11], anchor texts [3, 4, 5, 8], contents of globally link-related pages [5, 12, 13, 14], and contents of local surrounding pages [1, 9, 10]. All of these information sources except the last one are used to capture the features that are characteristic to the target pages, and are effective to emphasize highly probable pages. The last one, contrarily, is used to collect information dispersed over a logical page group, and is effective to gather potential pages comprehensively, but tends to increase noises. Since the comprehensiveness is a key factor for quality assurance of a web page collection, we mainly investigate to exploit the last one, i.e., the surrounding pages as the information sources.

Some works exploiting the surrounding pages first classify each page based only on its content and then combine the results based on other information sources such as link structure and directory structure [10]. However, when an entry page contains no textual information but hyperlinks, this approach will not work. Other works first cluster web pages based on local link structure and so on and then merge the score (or weight) of each word to generate a document vector [1, 9]. However, the effectiveness of this approach is limited, probably because it also merges many irrelevant words from the surrounding pages.

We also exploit the contents in surrounding pages considering local link structures, but with a different approach. In the rough filtering, for collecting as many target homepages as possible, page group models are applied for combining the local link structure among and the content of the pages in a logical page group, so that the homepages presented on a single page or on a set of pages that constitutes a logical page group can be gathered. In the accurate classification, the features on surrounding pages are partly merged then concatenated, and used in the classification all together, so that the contexts corresponding to the relative location are represented.

In addition, almost no prior works considered to assure the high quality required by practical applications. We approach to this problem by building a three-way classifier using a recall-assured classifier and a precision-assured classifier in combination.

3 Scheme of the Method

The scheme of the method is shown in Figure 1. It contains two step processes: the rough filtering and the accurate classification.

The rough filtering is for efficiently narrowing down the candidate page amount with a very high recall from the web. The input is all the web pages and the output is the candidate pages satisfying the required high recall. We set the performance required for the rough filtering as, for example, at least 98% and desirably 99%. Precision does not matter so much but a smaller amount of output pages is desirable under the condition on the recall.

Although we use a static web data corpus for the current work, the rough filtering method can be merged with a web crawler for real application. Then, it seems similar to focused crawling, but differs in several aspects. Firstly, focused crawlers predict the relevance of each page before fetching while the rough filtering does not. Consequently, it works only with comprehensive crawlers. Secondly, focused crawlers can handle one category at a time while the rough filtering can handle virtually as many categories as you like at a time.

The accurate classification is for accurately classifying the candidate pages output from the rough filtering into three classes: assured positive, assured negative, and uncertain. For example of the quality requirement, the recall should be at least 95% and the precision should be at least 99%.

Since even with the state-of-the-art classification technology, it is impossible to make the target data collections of the required quality solely by means of automatic processing. Therefore, human involvement is indispensable in overcoming the gap between the requirement and the technology. In order to assure the high performance, a relatively high computer processing cost is allowed for the accurate classification

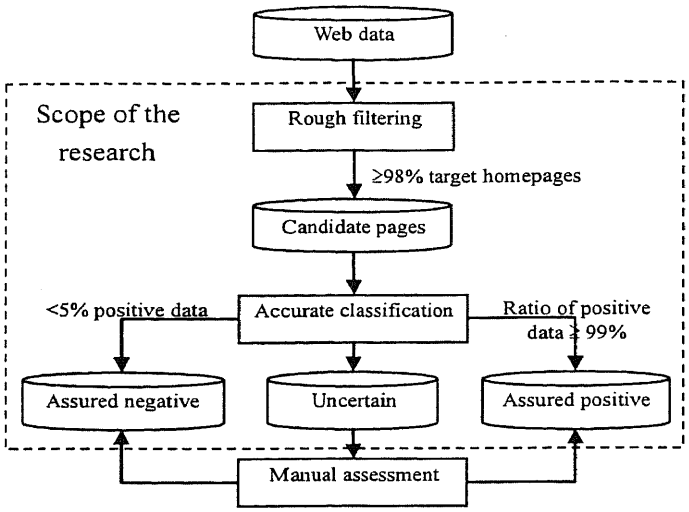

Fig. 1 Scheme of the method.

comparing to the rough filtering while the number of pages that need manual assessment should be reduced as many as possible.

4 The Rough Filtering

4.1 Structure of the Rough Filtering

The rough filtering uses property-based keyword lists and several kinds of page group models. Figure 2 illustrates the conceptual construction.

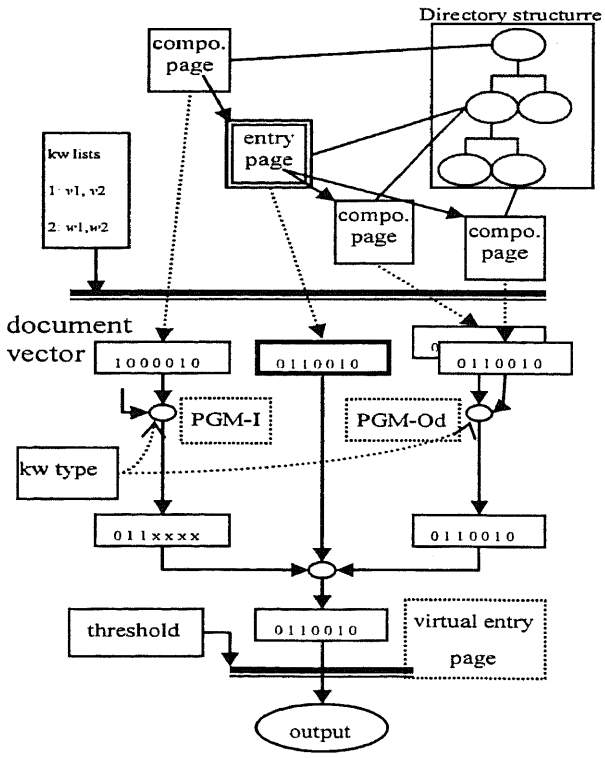

Fig. 2 Structure of the rough filtering.

Each web page is first mapped to a document vector consisting of binary values, each of which corresponds to a keyword list and represents if any of the keywords in the keyword list are present in the web page. Next, for each of the page group models, the document vectors are merged by making a logical sum of each vector element. In this process, only the elements corresponding to the keyword lists of suitable types for each page group model are considered (in the figure, ignored elements are indicated with 'x' at the output from PGM-I). They are further merged to the entry page's document vector to compose a final document vector. Here, a conceptual document represented by the final document vector is called a virtual entry page, and the process to merge the document vectors is called keyword propagation. Finally, scores of virtual entry pages are obtained by counting the number of 1's in the document vector, and those that scored more than or equal to a threshold score are output. The threshold score will be selected considering the evaluation results so that the recall satisfies the requirement, e.g., 99%, and the output amount is reasonable.

4.2 Property-based Keyword Lists

Although the styles and structures of homepages tend to differ greatly and the presentations are very diversity, they usually contain several basic information elements that are common to the homepages in the same category. Therefore, we introduce property-based keyword lists representing the common properties in the category, expecting that certain number (not necessarily all) of them are included in each target page or in its surrounding pages.

Even though some methods are available for automatically extracting content-based keywords, they are not applicable to extract property-based keyword lists where each of them contains a list of keywords grouped to the same property. Therefore we use an ad hoc method to create keyword lists for the present work and obtained 12 keyword lists containing 86 keywords for researchers' homepage category. We mainly use property-name-related terms. Property-value-related terms are used only when they can be enumerated within a small number; otherwise their maintenance would require a lot of effort.

Each of the keyword lists is then assigned a type either organization-related or non-organization-related. Keyword lists corresponding to the properties common to the members in the same organization are designated as organization-related, while keyword lists corresponding to individual researcher's properties are designated as non-organization-related. The types and meanings of these 12 keyword lists are listed in Table 1 along with some keyword examples. Note that the actual keywords are in Japanese.

Table 1 Property-based keyword lists and keyword samples

Type	Keyword lists	Keyword samples*
Non-organization-related	general word	research
	research topic	research topic, theme, etc.
	title	doctor, professor, etc.
	position	present position, duty, etc.
	history	biography, personal history, etc.
	achievement	paper, bibliography, etc.
	lecture	course, seminar, etc.
	academic society	academic society, regular member, etc.
Organization-related	major	major, specialty, research field, etc.
	member	staff, member, etc.
	organization	university, institute, school, etc.
	section	section, department, etc.

* Original keywords are in Japanese.

4.3 Page Group Models

Taking into account logical page group structure in the same site, we propose four simple page group models (PGMs) using (1) out-links to the same and lower directories, (2) out-links to the upper directories, (3) in-links from the same and upper directories, and (4) the directory entry pages in the same and upper directories in the URL directory path. The definitions of surrounding pages are listed in Table 2 and that of PGMs are listed in Table 3.

Table 2 Definitions of surrounding pages

Notations	Definition
r	current page
$P_{out}(r)$	set of pages linked from r in the same site (r's out-linked pages)
$P_{in}(r)$	set of pages linking to r in the same site (r's in-linked pages)
$P_{ent}(r[,s,l])$	set of directory entry pages in r's directory path from s to l level
$P_{same}(r)$	set of pages in the same directory as r
$P_{low}(r[,s,l])$	set of pages in the lower directory subtree of r from s to l level
$P_{upper}(r[,s,l])$	set of pages in the upper directory path of r from s to l level
$N_{lod}(r)$	number of links from page r to the pages in the same and lower directories of r

Note: The level of the same directory is defined as 0. s and l are options for specifying the ranges of the directory levels to propagate the keywords from. Default options means to use the pages in all levels of specified directories.

Table 3 Definitions for PGMs and parameters

Models		Description	Propagated pages	Parameters
SPM (baseline)		Single page model; no keyword propagation is used.	$\emptyset$	
SSM (baseline)		Reference page group model; all out-linked pages in the same site are used.	$P_{out}(r)$	
Simple PGM	Od(s,l)	A PGM based on out-links downward; out-linked pages in the URL directory subtree are used.	$P_{out}(r) \cap P_{low}(r, s, l)$	$s = 0, 1$; $l = s \,..\, 2$
	Ou(s,l)	A PGM based on out-links upward; out-linked pages in the directories included in the URL directory path are used.	$P_{out}(r) \cap P_{upper}(r, s, l)$	$s = 0, 1$; $l = s \,..\, 4$
	I(s,l)	A PGM based on in-links upward; in-linked pages in the directories included in the URL directory path are used.	$P_{in}(r) \cap P_{upper}(r, s, l)$	$s = 0, 1$; $l = s \,..\, 3$
	U(s,l)	A PGM based on directory entry pages; site top pages and entry pages of the directories in the URL directory path are used.	$P_{ent}(r, s, l)$	$s = 0, 1$; $l = s \,..\, 8$
Modified PGM	Od@θ	Od with additional conditions on the number of out-links; if there are too many out-links, Od is not used.	If $N_{Lod}(r) \leqq \theta$, same as Od; otherwise, same as SPM.	θ=5,10,20 ;
	Ou#, I#,U#	Ou, I and U, each propagating organization-related keywords only	Same as Ou, I and U for organization-related keywords; for others, same as SPM.	

Single page model (SPM) and single site model (SSM) are used as two baselines and are compared to the proposed PGMs in order to evaluate the effectiveness of the proposed PGMs. PGM-Od, PGM-Ou, PGM-I, and PGM-U are four simple PGMs. PGM-Od is intended to exploit all kinds of keywords in out-linked component pages in the lower levels of the directory subtree. PGM-Ou, PGM-I, and PGM-U are intended to exploit all kinds of keywords in component pages in the upper levels of the directory path: PGM-Ou for out-linked pages, PGM-I for in-linked pages, and PGM-U for directory entry pages, respectively.

Since simple PGMs usually propagate many keywords irrelevant to the researcher and consequently include many noisy pages, we propose modified PGMs to reduce such noises, whereas to keep useful keywords propagated.

PGM-Od@θ is a modified PGM derived from PGM-Od with the intention of excluding irrelevant pages, based on the observations that one of the noise sources is large groups of pages mutually linked within a directory, and that an entry page having many out-links always contains sufficient keywords in itself.

PGM-Ou#, PGM-I#, and PGM-U# are Modified PGMs derived from PGM-Ou, PGM-I, and PGM-U, respectively, with the intention of excluding irrelevant keywords based on the observation that non-organization-related keywords are not included in the upper directory hierarchies. Therefore only organization-related keywords are propagated with these PGMs. Since any single PGM can utilize only a part of the available component pages and can collect insufficient information, we combine all of the modified PGMs.

4.4 Experiments and Considerations

4.4.1 Data set

For the experiments, we used a corpus of 100GB web document data containing 11,038,720 web pages, NW100G-01, which was gathered from the '.jp' domain for WEB Tasks at the Third and Fourth NTCIR Workshops [15, 16].

A sample data set used for the rough filtering is prepared from NW100G-01. We first collected 113,380 pages containing some typical Japanese family names and randomly selected 11,338 pages, 10% from them (hereinafter we call this set of 11,338 pages as Jname data). Each of the pages was then manually assessed by the authors according to its content and, if necessary, the contents of the in/out-linked pages. Consequently, we obtained 426 positive samples and 10,912 negative samples.

We used another corpus for evaluating the effectiveness of the rough filtering. It is 1.36TB (1.5×10^{12} byte) web document data containing 95,870,352 web pages, NW1000G-04, which is created for the WEB Task at the fifth NTCIR Workshop [17].

4.4.2 Experiment results

The following experiments are based on NW100G-01. We select parameters for each PGM based on the following policy: if the difference of page amounts around the 99% recall area is small between two parameter sets, then the one that collects keywords from more pages should be selected.

First, we experimented on individual simple PGMs with typical parameters in order to understand their basic performances. The results show that all the simple PGMs deteriorate in their page amounts than SPM since a lot of noises are introduced by the keyword propagation, but with fewer noises than SSM.

Next, we experimented on modified PGM-Od, PGM-Ou, PGM-I, and PGM-U, and compared each of them to the corresponding simple PGM with typical parameters. The results of PGM-Od show that almost all non-organization-related keywords are collected from within the same directory. Since PGM-Od is the only PGM that propagates non-organization-related keywords, we selected s=0 although modified PGM-Od still collects a rather large amount of noise pages. Focusing on around the 99% recall area, the page amount increases by 80% over SPM with simple PGM-Od, whereas modified PGM-Od can reduce the increase down to 50%. The results of PGM-Ou, PGM-I, and PGM-U show that focusing on around the 99% recall area, although the page amount increases by 40% to 120% over SPM with each simple PGM, modified PGMs can reduce the increase to almost the same level as SPM.

Finally, we experimented on combinations of PGMs with several promising parameter sets. The results are shown in Figure 3 comparing to SPM and SSM plots. In the figure, the x-axis is the page amounts $n_c(i)$, namely, the number of pages in the corpus that scored at least i. The y-axis is the recall defined by $n_p(i) / N_p$, where N_p is the total number of positive sample data, $n_p(i)$ is the number of positive sample data that scored at least i ($1 \leq i \leq 12$). For each plot, the most up and right data corresponds to a threshold score 1, and every next one corresponds to a threshold score incremented by 1. In general, a higher recall and a less page amount indicate better performance; however, we put priority on recall.

Figure 3 shows the run results of top three best performed combinations of PGMs. We will refer to each of them hereinafter as follows:

PGM-C1: PGM-Od@5(0,2),Ou#(1,3),I#(0,3),U#(0,3)
PGM-C2: PGM-Od@10(0,2),Ou#(1,3),I#(0,3),U#(0,3)
PGM-C3: PGM-Od@20(0,2),Ou#(1,3),I#(0,3),U#(0,3)

Each of them uses all four modified PGMs with the same parameters except for θ of PGM-Od. As s= 0 is used for PGM-Od, s= 1 is selected for PGM-Ou. All the other parameters were eventually the same for all combinations.

The results show that even the best performed run PGM-C1 is inferior to SPM in all the recall ranges except for 100%. However, it is shown that the proposed method reduced the page amount to a certain degree despite its use of PGMs.

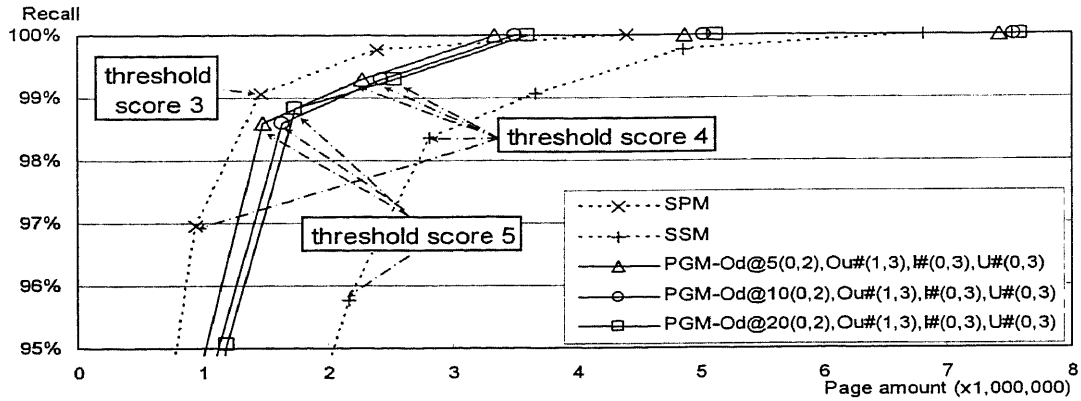

Fig. 3 Performance of typical PGM combinations.

4.4.3 Considerations

(1) Ability to find overlooked homepages

In order to evaluate the ability of the proposed method to find positive pages that were overlooked in the manual assessment, the pages that were contained in Jname data and scored less than 3 with SPM, but scored at least 4 with PGM-C3, were assessed and 13 new positive data were found. Then, for each of them, we checked the scores for SPM and PGM-C1 to -C3 respectively. The numbers of the data for each score are shown in Table 4. The result shows the ability of the proposed method to find positive pages that cannot be gathered by SPM even if we select the threshold score of 3 so that the recall is more than 99% for the manually assessed positive samples.

Table 4 Overlooked positive pages per score

		Score for SPM			
		0	**1**	**2**	**Total**
Score for PGM-C1 to C3	**4**	1	2	0	3
	5	2(1)	5	0	7(6)
	6-12	1	1	1	3
	Total	4(3)	8	1	13(12)

Note: Each cell indicates the number of positive page. The number in parentheses is that for PGM-C1 only.

Taking into account the new 13 positive data shown in Table 4, recalls of SPM at threshold scores of 2 and 3 should be corrected from 99.8% (425/426) to 97.0% (426/439) and from 99.1% (422/426) to 96.1% (422/439), respectively. By comparing these values with the recalls of the proposed methods at threshold scores of 4 and 5 respectively, it is obvious that the proposed methods outperform SPM with 5% significance. Furthermore, four of the positive pages cannot be gathered with SPM even if the threshold score is set to 1. This implies SPM can hardly achieve the goal recall at any feasible page amount.

Furthermore, a failure analysis on all the three pages that scored only 3 with PGM-C1 through -C3 revealed that they are in similar pattern and scored only 2 with SPM. Although they have hyperlinks to the researchers' personal homepages, our method cannot exploit them because they exist in separate sites. The facts support that for applications where only an informative homepage suffices when multiple homepages exist for a researcher, the proposed method must have worked if their personal homepages had been crawled.

Finally, as there are trade-offs between the recall and the page amount, it is difficult to say in general which of PGM-C1, -C2 and -C3 is the best. In order to guarantee that the overall recall will be more than 98% considering the confidence interval based on the positive sample data number, we should set the threshold score to 4. We will eventually select PGM-C2 as the most appropriate one for the current goal, because the recall at threshold score of 4 is the same for PGM-C2 and -C3.

(2) Applicability to a Larger Data Set

We applied the rough filtering to the larger data set NW1000G-04 with the procedure which is similar as that for NW100G-01. Approximate computational complexities of the overall processing cost for the rough filtering is $O(N \log N)$ where N is the number of the web pages in the corpus.

The same parameters of PGMs used for NW100G-01 are applied to NW1000G-04. The threshold number θ of out-link pages for PGM-Od is set as 20. The candidate pages are gathered with the threshold score 4. Table 5 presents the comparison of the experiment results.

Comparing the proportion of the pages output from the rough filtering, the experiment result shows that the output pages can be reduced more for the larger data set (less than 15% of the corpus). Therefore the rough filtering is not only applicable to but also more efficient for a larger data set. However, as we have not assessed the correctness of the output, stability of the accuracy is yet to be investigated.

Table 5 Comparison of pages output from the rough filtering for two data sets

Data set	Total amount	Output amount	Output proportion
NW100G-01	11,038,720	2,530,850	22.9%
NW1000G-04	95,870,352	14,128,826	14.7%

5 The Accurate Classification

5.1 Composition of the Accurate Classification

Figure 4 shows the composition of the proposed method (95% recall and 99% precision are the example quality requirement for illustration). We use two component classifiers to construct a three-way classifier. The recall-assured (precision-assured) classifier assures the target recall (precision) with the highest possible precision (recall).

The pages output from the rough filtering are first input to the recall-assured classifier and its negative predictions are classified to "assured negative". The rest are then input to the precision-assured classifier and its positive predictions are classified to "assured positive". The remaining pages are classified to "uncertain", which require manual assessment.

Since support vector machine (SVM) is shown to be efficient and effective for text classification, we use SVM^{light} package [18] by Joachims with linear kernel in the current work, tuning with its options c (tradeoff between training error and margin) and j (cost-factor by which training error on positive examples out-weight errors on negative examples). For all the experiments, the performance of each classifier composed by a feature set is evaluated by precision, recall, or F-measure which are defined as:

Precision = #correct positive predictions / #positive predictions
Recall = #correct positive predictions / #positive samples
*F-measure = 2*precision*recall / (precision+recall)*

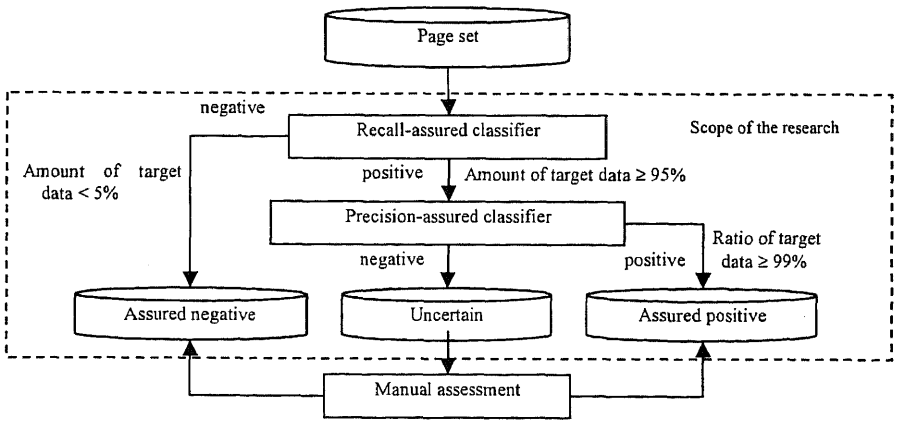

Fig. 4 Composition of the accurate classification.

5. 2 Surrounding Page Group and Feature Set

When a page (current page) is given, its surrounding pages are categorized to groups $G_{c,l}$ based on connection types c (in-link (*in*), out-link (*out*), and directory entry (*ent*)) and URL hierarchy levels l (*same*, *upper*, and *lower*) relative to the current page. The current page constitutes an independent group G_{cur} and all defined surrounding page groups are shown in Table 6. Each group of $G_{c,l}$ has its own potential meaning in a logical page group. For example, $G_{in,low}$ consists of in-link pages in lower directories which might represent component pages having back link to the entry page, and $G_{ent,upper}$ consists of directory entry pages in upper directories which might represent entry pages of the organization the researcher belongs to.

We use textual features $f_{t,v}(g,w_t)$, where t indicates a text type *plain* (plain-text-based) or *tagged* (tagged-text-based), v indicates a value type *binary* or *real*, g denotes a surrounding page group, and $w_t \in W_t$ denotes a feature word. Then, corresponding to the g of each $G_{c,l}$, a feature subset $f_{t,v}(g,w_t)$ is generated and feature sets are further composed by concatenating one or more feature subsets $F_{t,v}(g) = \{ f_{t,v}(g,w_t) \mid w_t \in W_t \}$. Figure 5 illustrates the way to compose the feature sets. For instance, feature set "u-1" shown in Subsection 5.4 is composed by concatenating feature subsets on pages in G_{cur} and $G_{*,upper}$ (surrounding page groups of upper hierarchy level) and feature set "o-i-e-1" is composed by concatenating feature subsets on pages in G_{cur} and $G_{*,*}$ (all surrounding page groups).

Table 6 Groups of surrounding pages

	r	$P_{ent}(r)$	$P_{in}(r)$	$P_{ent}(r)$	Merged
$P_{same}(r)$	G_{cur}	$G_{out,same}$	$G_{in,same}$	$G_{ent,same}$	$G_{all,same}$
$P_{low}(r)$		$G_{out,low}$	$G_{in,low}$		$G_{all,low}$
$P_{upper}(r)$		$G_{out,upper}$	$G_{in,upper}$	$G_{ent,upper}$	$G_{all,upper}$
Merged		$G_{out,all}$	$G_{in,all}$	$G_{ent,all}$	$G_{all,all}$

5.3 Text Type, Feature Word, and Value Type

We use two kinds of textual features. Plain-text-based features $F_{plain,*}(*)$ are extracted from textual content excluding tags, scripts, comments, etc. We use *Chasen* [19] for Japanese and *Rainbow* [20] for English to tokenize page contents. Top 2,000 words are selected as feature words W_{plain} based on mutual information.

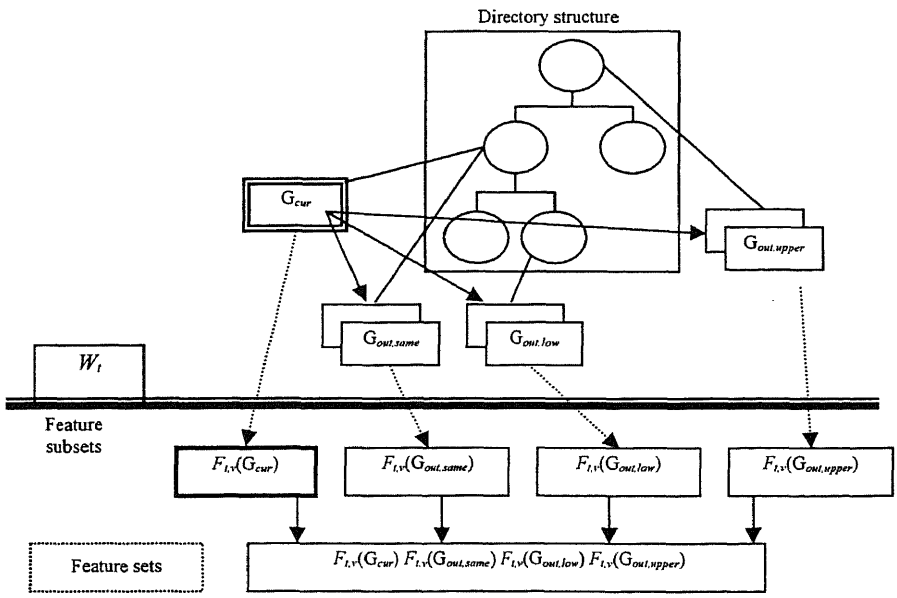

Fig. 5 Composition of feature sets.

Tagged-text-based features $F_{tagged,*}(*)$ are extracted from text segments "*text*" that match either ">*text*<" or "<img...alt= "*text*"...>" and that are not more than 16 bytes long omitting spaces in Japanese case and 4 words in English case. We again use *Chasen* and *Rainbow* to tokenize the extracted text segments. The obtained words with less than 1% file frequency in Japanese and all the obtained words in English are used as feature words W_{tagged} since all of them are considered as property words. In the experiments, a feature set is composed by feature words either *plain* alone or *plain* and *tagged* together, with the latter indicated by the suffix "_tag" of the run name.

A *binary* value $f_{*,binary}(g,w_t)$ represents the presence of w_t in g. A *real* value $f_{*,real}(g,w_t)$ represents proportion of pages containing w_t within g. The *real* value is tested to see if feature word distribution within surrounding page groups is informative. The two value types are used exclusively for composing a feature set. Using *real* value type is indicated by the suffix "_real" of the run name.

5.4 Experiments and Considerations

5.4.1 Experiments using Web->Kb data set

(1) Web->Kb data set

We used Web->Kb data set (in English) for testing the effectiveness of proposed features. It is provided by the World Wide Knowledge Base (Web->Kb) Project at the CMU text learning group and is commonly used as a test collection for the web page classification task. It contains 8,282 pages collected from computer science departments of 4 universities and other miscellaneous universities. All data are classified into 7 categories and 4 categories, student, faculty, course, and project, are used in the experiment.

(2) Experiment process

As recommended by the project, we used *Leave-one-university-out* cross-validation method. The pages of miscellaneous universities are always used as training data. For comparison, a feature set composed by $F_{plain,binary}(G_{cur})$ only is used as the baseline.

The features were extracted with the method shown in Subsection 5.3. Around 600 tag-based feature words were extracted for each category of training-testing data pair.

(3) Experiment results

The experiment results of well performed feature sets on Web->Kb data set are shown in Table 7. The performance of each classifier composed by a feature set is evaluated by the best F-measure tuned with c and j options. The overall experiment results show that, tag-based features are consistently effective and the differences caused by the feature values, binary or real, are negligible. o-i-e-1_tag and o-i-e-1_tag_real performed the best and u-1_tag also performed rather well considering its relatively simple feature composition.

Table 7 Classification results of Web->Kb data set (in percentage of F-measure)

Classifier	course	faculty	project	student	Macro(4)	Macro(3)
baseline	68.41	76.01	39.62	74.95	64.75	73.12
o-i-e-1	76.97	78.27	53.30	72.53	70.27	75.92
u-1_tag	75.13	77.37	41.49	71.51	66.38	74.67
o-i-e-1_tag	77.82	79.60	59.49	74.53	72.86	77.32
o-i-e-1_tag_real	77.09	79.35	57.24	75.04	72.18	77.16

(4) Comparison to prior works

We compared the experiment results with proposed method to prior works which used Web->Kb data set too in Table 8. The results show that our method out-performed all the seven prior works based on macro-averaged F-measure of all the four categories (Macro(4)) and is a little inferior to only 1 of the seven prior works on macro-averaged F-measure of course, faculty, and student categories (Macro(3)). Our method out-performed 10 out of 12 on per-category basis (F-measures of the individual categories are not available for 4 of the prior works).

Table 8 Performance comparison to prior works (in percentage of F-measure)

method	course	faculty	project	student	Macro(4)	Macro(3)
o-i-e-1_tag	77.8	79.6	59.5	74.5	**72.9**	**77.3**
o-i-e-1_tag_real	77.1	79.4	57.2	75.0	72.2	77.2
FOIL(Linked Names)[11]					62.9	
FOIL(Tagged Words)[11]					59.1	
SVM(TA)[3]	68.2	65.9	32.5	73.0	59.9	69.0
SVM-FST(XATU)[4]	60.9	40.9	66.5	25.3	48.4	42.4
ME(TU)[6]					62.7	
SVM-iWUM(α=1)[10]	54.7	87.6	17.1	95.8	63.8	79.4
GE-CKO(FC5)[5]						76.5

5.4.2 Experiments using NW100GB-01 sample data set

(1) NW100G-01 sample data set

We prepared the sample data set from NW100G-01. Firstly 20,846 pages (1%) were randomly sampled from the rough filtering output, and then each page was manually assessed based on its content (and the content of its surrounding pages, if necessary). Consequently, we obtained 480 positive samples and 20,366 negative samples. We use the former and the 426 positive samples used in the rough filtering experiment together (906 in all) as the positive sample data, and the latter as the negative sample data.

(2) Experiment process

Five-fold cross validation is adopted for all experiments on NW100G-01 sample data set. For comparison, a feature set composed by $F_{plain,\,binary}(G_{cur})$ only is used as the baseline.

First we experimented with feature sets composed by feature subsets $F_{plain,binary}(g)$ with all possible combinations of g's. Among them, we selected several feature sets that performed relatively high recall (precision) at high precision (recall) area. For the selected feature sets, $F_{tagged,binary}(g)$ in addition and $F_{*,real}(g)$ instead are applied to them too.

(3) Experiment results

Overall performances of typical classifiers on NW100G-01 sample data set are shown in Figure 6, and their details in high precision/recall areas are shown in Figure 7 and Figure 8, respectively. Each curve is drawn by connecting results of well-performing tuning parameters. The performance measures of well-performing feature sets and the baseline are shown in Table 9. The overall experiment results show that, in the high precision area, the best and the second best performing feature sets o-i-e-1_tag_real and u-1_tag slightly outperform the baseline but the effectiveness is unclear. In contrast, in the middle to high recall area, o-i-e-1_tag_real evidently outperforms the baseline and u-1_tag also performs rather well.

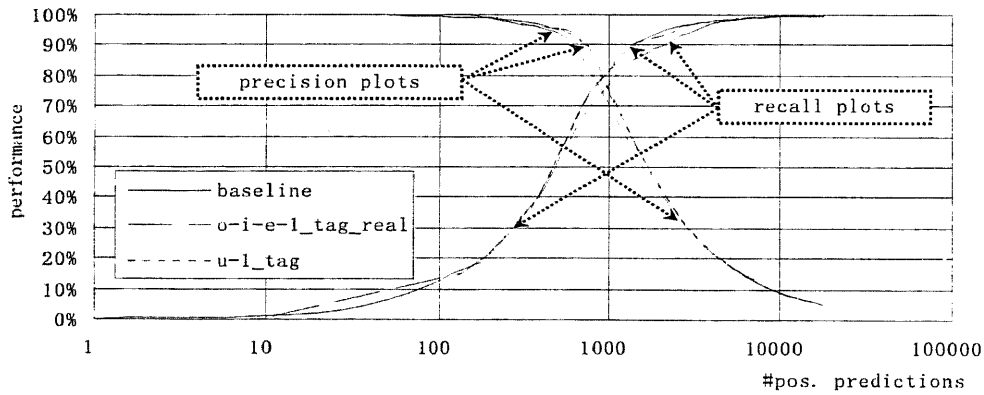

Fig. 6 Overall performances of NW100G-01 sample data set.

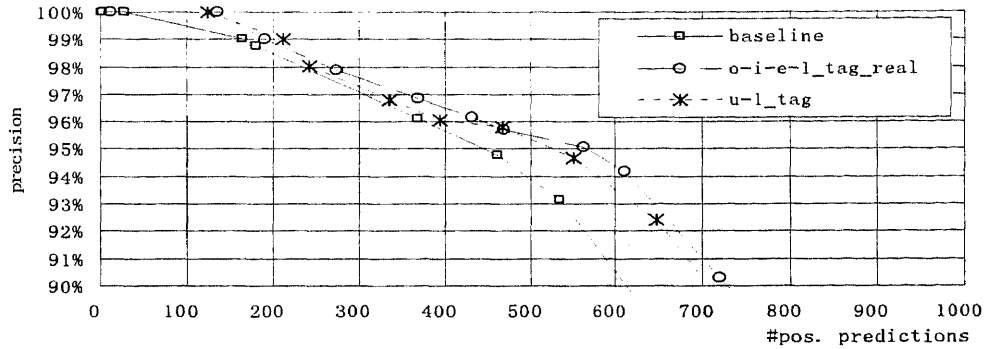

Fig. 7 Precision at high precision area of NW100GB-01 sample data set.

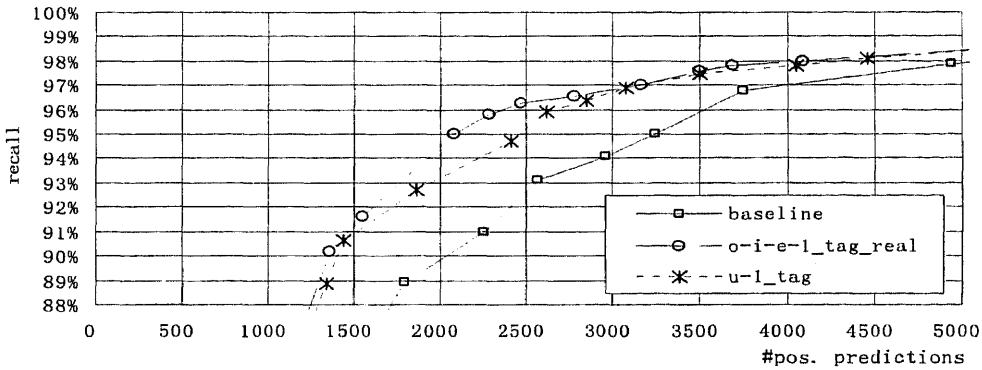

Fig. 8 Recall at high recall area of NW100GB-01 sample data set.

Table 9 Performance of well performed feature sets (in percentage of F-measure)

Feature set	Best F-measure	Recall at 99% precision	Precision at 95% recall
o-i-e-l_tag_real	88.65	20.86	41.33
u-l_tag	88.58	23.18	33.22
baseline	83.26	17.98	26.49

5.4.3 Considerations

(1) Effectiveness of the proposed features

Based on the experiment results on NW100G-01 sample data set and Web->Kb data set, the proposed features all together are shown to be effective for the general performance (in F-measure). In addition, the experiment results on Web->Kb data set compared to other prior works (shown in Table 8) not only show the effectiveness of the proposed features but also show the possibility the proposed method is applicable to other research-related homepage categories and/or in other language than Japanese.

Besides, the experiment results on NW100G-01 sample data set show the proposed features are effective for the performance of precision/recall-assured classifiers too. Surrounding page groups are generally effective but their contributions vary.

For precision-assured classifiers, the upper hierarchy page groups ($G_{*,upper}$) contribute the most to the recall. It probably indicates that such pages provide contextual information, e.g., organization names and research fields, which is lacking in the current page itself but is very important for classifying them with very high confidence. For recall-assured classifiers, all surrounding page groups ($G_{*,*}$) contribute to the precision notably, despite of their noisy natures.

Other experiment results not presented here have shown the followings. (1) If we group the surrounding pages based either on connection types or on hierarchy levels, much poorer performance would be obtained. (2) Adding tagged-text-based features consistently gained performance. It can be interpreted that many

noisy information from the surrounding pages are suppressed by the tagged-text-based features and consequently their useful information can be exploited. (3) Effect of value types varies depending on feature sets and its gain is marginal.

(2) Reduction of manual assessment

In order to know the reduction of the pages requiring manual assessment (i.e., the pages classified as uncertain) by using the proposed method, we compare two compositions of three-way classifiers. Table 10 shows estimated page numbers of classification output from NW100G-01 at three different quality requirements for two three-way classifiers, one using baseline and the other using the best performed feature set, i.e. o-i-e-1_tag_real, as both recall/precision-assured classifiers. Comparing the "uncertain" class sizes, o-i-e-1_tag_real significantly reduces the amount of pages requiring manual assessment, especially when the required quality is relaxed.

Table 10 Estimated page numbers of classification output from the corpus

Required quality	baseline			o-i-e-1_tag_real			reduction ratio
precision./ recall	assured positive	uncertain (Nb)	assured negative	assured positive	uncertain (No)	assured negative	No/Nb
99.5% / 98%	3800	461832	1618988	9206	358207	1717187	77.6%
99% / 95%	6163	274524	1803913	11251	156782	1916567	57.1%
98% / 90%	11116	155418	1918066	15503	81157	1987940	52.2%

6 Conclusions and future works

In this paper, we proposed a realistic framework to assure the quality of the web page collection with two step processes: the rough filtering and the accurate classification. In both processes, we introduced an idea of local page group structure and demonstrated its effective uses for filtering and classifying web pages, where researchers' homepages are used as an example.

In the rough filtering, we described the method for comprehensively gathering all probable researchers' homepages from the web within as few noise pages as possible. We proposed a method of using property-based keyword lists combined with four page group models. Two original key techniques were used to reduce irrelevant keywords to be propagated by exploiting the mutual relations between the content and the structures among pages in a logical page group: out-link-number-based and keyword-list-type-based keyword propagation.

In the accurate classification, we proposed a web page classification method for building a high quality homepage collection using support vector machine (SVM) with textual features obtained from each page and its surrounding pages. The surrounding pages are grouped based on the relative location considering the connection types and the relative URL hierarchy, and an independent feature subset is generated from each group. Using the feature set composed by concatenating the feature subsets which is our original key technique, we have achieved an evident improvement of classification performance. Furthermore, by using a recall-assured classifier and a precision-assured classifier in combination, we presented the method for accurately classifying input data to assured positive, assured negative and uncertain classes for assuring given precision and recall, so that only the "uncertain" output should be manually assessed. Applying the proposed features, we have shown the amount of the "uncertain" output is significantly reduced.

Although we have not applied our method to other categories, we expect our method is effective to many other categories, such as shopping, product catalogs, and so on. In terms of information service, the high-quality collections built with our method will be applicable to various domain specific search engine with guaranteed high quality.

Even though our method is efficient and effective to a certain degree to fulfill the objectives of the research, it is not satisfactory yet. Hence, we will continue the work mainly on the following issues:

For the rough filtering, we will try to find a systematic way for extracting the property-based keywords and the property set and to combine individual keyword lists with other possible keyword types in a systematic way.

For the accurate classification, we will try to exploit various features on other promising clues, such as file types of link target, anchor text, or page tag structure, etc. We will further investigate the way to estimate the likelihood of the component pages and to introduce it to the current method.

In conclusion, to tackle the diversity of web data by exploiting their rich web-based features pursuing a very high performance with less processing cost is a challenging problem. The method presented in this paper is considered to give a general framework for solving the same kind of problems and we hope that the method will benefit and contribute to the related research on web information utilization.

Acknowledgements

We used NW100G-01 and NW1000G-04 document data sets under permission from the National Institute of Informatics. We would like to thank Professors Akiko Aizawa and Atsuhiro Takasu of NII for their precious advice.

References

1. Y. Wang and K. Oyama. Combining page group structure and content for roughly filtering researchers' homepages with high recall. IPSJ Transactions on Databases, Vol.47, No.SIG 8 (TOD 30), (2006) 11-23.
2. S. Chakrabarti. Data mining for hypertext: a tutorial survey. ACM SIGKDD Explorations, Vol. 1, No. 2, (2000) 1-11.
3. A. Sun, E.-P. Lim, and W.-K. Ng. Web classification using support vector machine. In Proc. of the fourth international workshop on web information and data management, ACM Press, (2002) 96-99, McLean, Virginia, USA.
4. M.-Y. Kan. Web Page Categorization without the Web Page. In Proc. of 13th World Wide Web Conference (WWW2004), New York, NY, USA, May 17-22, (2004).
5. J. Sun, B. Zhang, Z. Chen, Y. Lu, C. Shi, and W. Ma. GE-CKO: A Method to Optimize Composite Kernels for Web Page Classification. In Proc. of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence (WI2004), (2004) 299-306, Beijing, China.
6. M.-Y. Kan and H.O.N. Thi. Fast webpage classification using URL features. CIKM'05, (2005) 325-326, Bremen, Germany.
7. L. K. Shih and D. R. Karger. Using URLs and table layout for web classification tasks. WWW2004, (2004) 193-202, New York, NY, USA.
8. M. Craven and S. Slattery. Relational Learning with Statistical Predicate Invention: Better Models for Hypertext. Machine Learning, Vol. 43(1-2), (2001) 97-119.
9. T. Masada, A. Takasu, and J. Adachi. Improving web search performance with hyperlink information. IPSJ Transactions on Databases, Vol.46, No.8, (2005) 48-59.
10. A. Sun and E.-P. Lim. Web unit mining: finding and classifying subgraphs of web pages. In Proc. of International Conference on Information and Knowledge Management (CIKM2003), (2003) 108-115, New Orleans, Louisiana, USA.
11. Y. Yang, S. Slattery, and R. Ghani. A Study of Approaches to Hypertext Categorization. Intelligent Information Systems, volume 18, (2002) 219-241. Kluwer Academic Press.
12. S. Chakrabarti, B. Dom, and P. Indyk. Enhanced hypertext categorization using hyperlinks. In Proc. of Int. Conf. Management of Data (SIGMOD '98), (1998) 307-318, Seattle, WA, USA.
13. M. Chau. Applying web analysis in web page filtering. In Proc. of the ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'04), (2004) 376, Tucson, Arizona, USA.
14. E. J. Glover, K. Tsioutsiouliklis, S. Lawrence, D. M. Pen- nock, and G. W. Flake. Using web structure for classifying and describing web pages. In Proc. of the 11th International World Wide Web Conference, (2002) 562-569, Honolulu, Hawaii, USA.
15. K. Eguchi, K. Oyama, E. Ishida, N. Kando, and K. Kuriyama. Overview of the web retrieval task at the third NTCIR Workshop. NII Technical Report, No.NII-2003-002E, NII, (2003).
16. K. Eguchi, K. Oyama, E. Ishida, N. Kando, and K. Kuriyama. Evaluation methods for web retrieval tasks considering hyperlink structure. IEICE Transactions on Information and Systems, Vol. E86-D, No. 9, (2003) 1804-1813.
17. K. Oyama, M. Takaku, H. Ishikawa, A. Aizawa, and H. Hamana. Overview of the NTCIR-5 WEB Navigational Retrieval Subtask-2. In Proc. of NTCIR-5 Workshop Meeting, Tokyo, Japan, Dec. 6-9, (2005).
18. SVM^{light} Support Vector Machine. http://svmlight.joachims.org/.
19. *Chasen*. http://chasen.naist.jp/hiki/ChaSen/.
20. *Rainbow*. http://www.cs.cmu.edu/afs/cs/project/theo-11/www/naive-bayes/gentle_intro.html.