Adaptive Content-based Navigation Generating System: Data Mining on Unorganized Multi Web Resources

Diana Purwitasari[1], Yasuhisa Okazaki[2], and Kenzi Watanabe[2]
[1]Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[2]Department of Information Science, Saga University, Saga, Japan

***Abstract*— Rapid growth of the Internet makes Web-based applications becomes a new means to learn. Application for learning takes into account of creating navigation as an important task to help users in understanding structured idea of learning topics within its materials collection. Let multi resources become the collection of a learning application. Then the arising problem is how to structure such abundance resources as navigation in the application that could satisfy different interests of users.**

Our adaptive content-based navigation generating system provides a domain of subjects called topics extracted from the collection. System automatically produces an organized structure of topics offering users' guidance for learning. Since the collection consists of Web pages, system will equally exploit information of contents and hyperlinks with the aim of generating navigation. When context interests of users changes like in a time user clicks a topic to know more about it, content-based navigation will adapt. System exploits user model because of that adaptation.

***Index Terms*— navigation generating system, text mining, unorganized web resources**

I. INTRODUCTION

WITH such availability of abundance resources in the Internet, there are lots of learning site confined to certain coverage knowledge area. Some subjects in one knowledge area might be overlap to other coverage. Ones could want to put together existing learning materials from multi resources in order to develop learning system that accommodates different subjects of users' interests. Scopes of subjects are hidden topics in the whole system collection of learning materials gathered from multi resources.

Some users might not yet have enough structured idea of learning topics. Navigation, an organized structure from a collection of learning materials, offers guidance for learning to help users. When dealing with such abundance resources, navigation map of learning topics would be out of date if it is manually constructed. It is easier to maintain structure of multi resources in the manner of hand over navigation creating tasks to a system.

Content-based navigation is defined as a sequence list of topics and sub topics hierarchically structured from a collection of documents. In our system we use data mining techniques and then followed by hypergraph partitioning to extract topics. Mapping of topics is considered as hypergraph construction with vertices representing words and edges representing strength relation between words. Extracted topics will be restructured to produce a hierarchical topics list using modified agglomerative clustering. System utilizes information of contents and hyperlinks in the materials collection of Web pages to provide the list.

Implementation at educational fields, which particularly utilize services on the Web, refers the term of adaptive as information filtering. That is to say as finding relevant items to user interests in large collection of document. Therefore content-based navigation adapts to users when context of users' interests changes like in a time user clicks a topic to know more about it.

II. OUR PREVIOUS WORKS AND RELATED WORKS

A variety of educational resources available to users on the Web for almost every domain is changing rapidly. However, the abundance of resources has created a problem of finding and organizing resources that match individual goals, interests, and current knowledge of users. Web-based systems for learning with the beneficial of adaptivity and intelligence provide an alternative to the traditional "just-put-it-on-the-Web" approach. That makes personalized access for users becomes the issue. Web-based learning systems should behave differently for different users to provide the adaptation effect. So far, the only techniques that demonstrate a good ability to provide personalized access to information in the learning context are

D. Purwitasari is with the Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, e-mail: diana@if.its.ac.id.

Y. Okazaki and K. Watanabe are with Department of Information Science, Saga University, Saga, Japan.

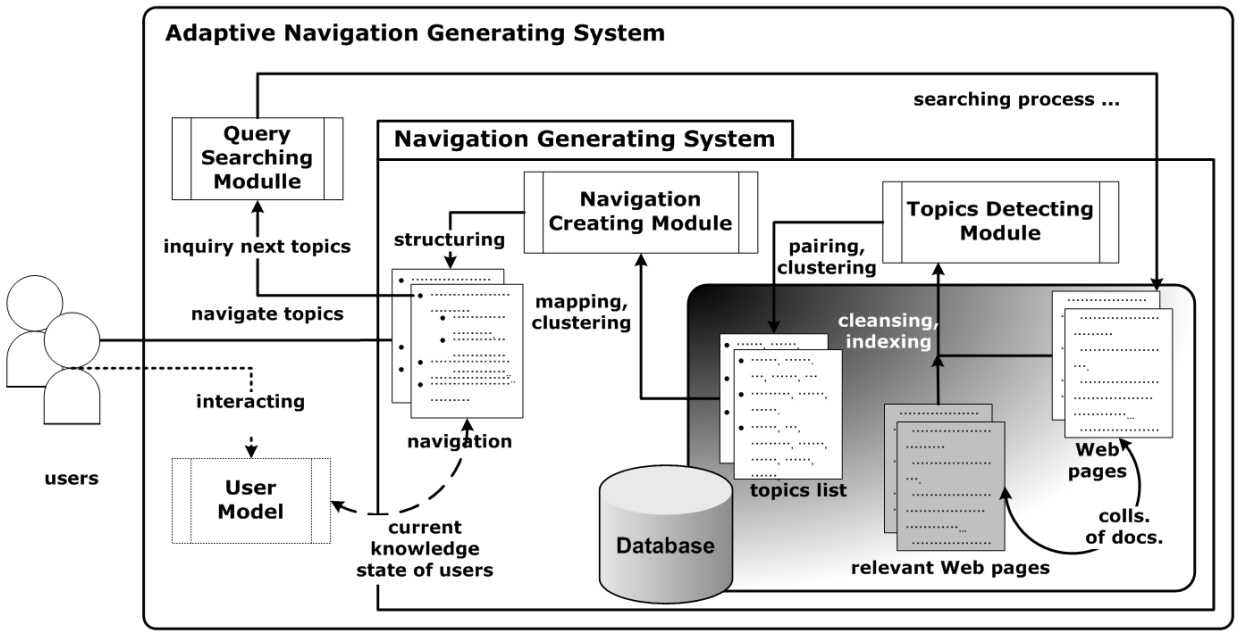

Fig. 1. Framework of adaptive content-based navigation generating system.

adaptive navigation support [1] such as curriculum sequencing. The goal is to provide a user with "optimal path" through the materials that is the most suitable individually planned sequence of topics for learning. This becomes very important due to its ability to guide users through the volume of available resources. Sequencing could be implemented in the form of list of some recommended links as a suggested learning path.

We design generating system which produces appropriate learning path with current interest of users. A document of Web page is representation form of at least an identified topic in learning path. On account of users want to browse more of a topic, system translates users' clicked action into inquiry using feature terms within the Web page as search keywords of selected topic. Our scoring schema exploits information of contents and hyperlinks in Web pages [2].

Many research efforts, which influence our framework to generate content-based navigation, have been engaged to bring structured representation to a large collection of documents in the Web. To fulfill the function there are two main approaches: (i) organize documents in a domain of subjects [3] [4] [5] [6] and (ii) provide scenarios of topics for learning particular subjects [7] [8].

We do thorough study literature to propose a framework that makes the best use of both models [9]. Then prototype implementation of the framework shows system uniqueness compare to works of previously mentioned researches. With data mining techniques on topic identification [3], we adapt the techniques and then employ similarity function for merging topics to other tasks like structuring and representing document into topics within content-based navigation. Documents may have membership in several domains of subjects. Though fuzzy clustering might be more suitable in partition collection of subjects-overlapped documents [4], we show even agglomerative clustering can be applied. We consent in exploiting information of hyperlinks as well [5] [6] to generate navigation of learning path because documents in the materials collection are Web pages.

Evaluations are done to verify validity of selected methods in the framework [10]. Some evaluations concerns about preparing Web contents that should be sufficient for a collection thus the system could suitably generate navigation. Another evaluation is related to comparing results of some distance measures for partitioning criteria on agglomerative clustering. Other evaluation observes generating navigation from contents of some documents that close to user's current interest of topic. The documents are retrieved with our scoring model [2].

After observing number of evaluations, the proposed framework is implemented [11]. However it is still a model system which generates a content-based navigation. An adaptive effect to generate and re-generate navigation that supports statement to produce appropriate learning path with current interest of users is emphasized in current work.

III. CONTENT-BASED NAVIGATION GENERATING SYSTEM

Given a collection of documents, system outlines learning topics then provides learning scenarios to offer guidance for users. Initially it extracts topics which are frequently discussed in the collection with data mining techniques (Topics Detecting Module, Fig.1), then clusters extracted topics in order to hierarchically produce a sequence list of topics (Navigation Creating Module, Fig.1)

A. Overview

Basic cognitive process of topic identification is that there exist sets of common words (*frequent itemsets*) for each topic. Frequent itemsets are mapped into hypergraph with vertices represent words and edges represent strength relation between words (Topics Detecting Module, Fig.1). To extract topics means to partition hypergraph into sub-graphs. Then extracted topics will be restructured to produce a hierarchical list of topics-subtopics using agglomerative clustering with some adjustments. This hierarchical list is supposed to give an implicit path as kind of learning direction for users (Navigation Creating Module, Fig.1).

When users click in one topic, system makes an inquiry using feature terms within Web page as search keywords of selected topic. Inquiry results retrieved with our scoring scheme [2] become new resources to produce next sequence of topics (Query Searching Module, Fig.1). This module becomes addendum of adaptive effect in adaptive content-based navigation generating system.

B. Topics Detecting Module

If important terms (frequent itemsets) in the collection are mapped into graph of word vertices, sub graphs indicate implicit topics. Hypergraph is a graph in which its edge can connect more than two vertices. This module partitions hypergraph to single out hyperedges for identifying mostly discussed topics. *shmetis* library in *hMetis* tool is executed to do hypergraph partitioning[12]. Based on defined writing rule by *shmetis* for input-output files, we provide ways of encoding frequent 2-itemsets of hypergraph into input file and decoding output file into list of unnamed topics consisting common words of each topic.

Topics Detecting Module begins with preprocessing collection [13] which includes indexing, removing stop words, stemming, and TF-IDF (term frequency - inverse document frequency) weighting in order to retrieve candidates of frequent itemsets. Data mining lists frequent itemsets that satisfying some minimum values of support and confidence filters. We introduce usage of other filters to set apart significant frequent 2-itemsets. First is indispensable feature value of TF-IDF term weight (Eq.1), and second is our own weighting schema of term entropy (Eq.2).

$$\omega_{d,t} = \frac{tf_{d,t}}{\max_i tf_{d,t}} \cdot \log \frac{N}{n_t} \geq \delta_w \quad \text{.....(1)}$$

Let the collection D of N Web pages, $|D| = N$, shows n_t as number of documents containing term t. Then a weight $\omega_{d,t}$ is assigned for each term t in a document d that depends on the number of term occurrences in the document, $tf_{d,t}$, normalized by frequency value of the highest term occurrences. Term weight $\omega_{d,t}$ is attenuated with an effect of $idf_t = \log N/n_t$ if the term occurs too often in the collection. For TF-IDF filter we do not consider values of term weight $\omega_{d,t}$ less than δ_w.

TABLE I
A CONTINGENCY TABLE TO HELP DETERMINING IMPORTANCE STATE OF TERMS IN A DOCUMENT

	belongs to C_D	belongs to C_{NOT_D}	
from set S_{lt}	a	b	$\|S_{lt}\| = a + b$
from set S_{gt}	c	d	$\|S_{gt}\| = c + d$

Entropy value measures uncertainty state. We consider entropy value of a term reflects effectiveness of the term in identifying certain document to others. This reasoning is derived from selection of most useful attribute to test at each node in decision tree algorithm [14]. For term entropy calculation we assume that there are only two classes as target classification:

(i) whether the term is one of important terms in currently observed document or

(ii) in the contrary that the term is important for any other documents except the currently observed one.

Attribute for classification of a term belongs to (i) or (ii) is test condition whether frequency of the term in currently observed document (*term frequency*) will be less or more than average value of term frequency from all documents in the collection.

In a collection D, for a document d let term t is classified into two classes:

(i) important terms of document d, C_d, or

(ii) important terms in other documents, C_{not_d}.

The state whether $tf_{d,t}$ value is less or greater than average number of term occurrences t in any document within collection $avg(tf_{d,t})_{d \in \{1...\}}$ becomes attribute for classifying.

For each calculation of term entropy value $ent_{d,t}$ in document d, we assume that there are only two classes as target classification. Term entropy value $ent_{d,t}$ is defined as:

$$ent_{d,t} = \sum \frac{|S_n|}{N} ent\_p_{d,t}(S_x) \geq \delta_e \quad \text{.....(2)}$$

with S_x can be set of documents where term occurrences $tf_{d,t}$ in current document d is less, S_{lt}, or greater, S_{gt}, than average number $avg(tf_{d,t})$; $S_x \in \{S_{lt}, S_{gt}\}$, and $D = \{S_{lt} \cap S_{gt}\}$ (Table 1 depicts contingency table to help determining importance state of terms in a document). For entropy filter we do not consider values of term entropy less than δ_e.

Given a set S_x, containing only positive and negative documents of 2 class problem C_d and C_{not_d}, entropy of set S_x relative to this simple, binary classification is defined as:

$$ent\_p_{d,t}(S_x) = -(p_p \log_2 p_p + p_n \log_2 p_n) \quad \text{.....(3)}$$

where p_p is proportion of documents with term t belongs to C_d in S_x and p_n is proportion of documents belongs to C_{not_d}.

We list frequent itemsets that satisfy data mining filters, which are minimum support (Eq.4) and minimum confidence (Eq.5) [15]. Support s value shows a percent value of n documents that must contain terms t_i and t_j, pair of common words occurs together in

multiple documents. The support filter will ignore any 2-itemsets of words that occur in few documents to ensure the ones that often occur are worthy of attention.

$$s_{i,j} = p(t_i, t_j) \geq \delta_s \quad \text{.....(4)}$$

While confidence *c*, also called as mutual information, measures dependence or correlation between occurences of terms t_i and t_j. The confidence filter makes further analysis to find whether the presence of a 2-itemsets that *often* occurs (read: sufficient to the support filter) is strongly significant to become a potential frequent itemsets.

$$c_{i,j} = \log_2 \frac{p(t_i, t_j)}{p(t_i) \times p(t_j)} \geq \delta_c \quad \text{.....(5)}$$

Note, $p(x)$ is defined as a probability function of x, i.e. $p(t) = n_t/N$, while $p(x, y)$ is a joint probability function of x and y.

C. Navigation Creating Module

To know relations between topics-subtopics, agglomerative clustering is used to organize topics in nested groups of merged topics [3]. Clustering initialization uses extracted topics from hypergraph partitioning as cluster seeds. In the first iteration a cluster contains single topic but for the next a cluster could contain more than one merged topics. Merging some topics in a cluster could result well comprehensive topic or not is an appraising question. Thus we judge within distances on members in the same cluster should be minimized while between distances on different clusters be maximized. The smaller value variance ratio of between and within distance shows, the better clustering results are. If certain iteration has large value of variance ratio though it inclines smaller at previous iterations, then clustering will cut-off tree of topics-subtopics.

Our assumptions during clustering process are that each document can only belong to exactly one cluster of topics and any cluster without document members will be removed. We use extracted topics as cluster seeds in initialization. Because of the assumptions there is possibility that some initial clusters do not have any document members and will be left out during tree construction.

Our proximity matrix *P* is a matrix whose ij^{th} entry denotes the similarity between topics in i^{th} and j^{th} clusters. Before merging topic *a*, tpc_a, and topic *b*, tpc_b, we recalculate similarity between any document *d* to soon-to-be-merged topics based on their common words ($sim(d, tpc_a)$, and $sim(d, tpc_b)$).

It means that there are words that not only frequently found in documents close to tpc_a but also exist in documents of tpc_b. The similarity of document-to-topic is derived from TFIDF formula in term weighting process (Eq.6) [3]. Accumulation of document similarities value with every topics pair is applied in a mutual information style to update proximity matrix of topic clusters *in each iteration time* clustering. Eq.7 defines sim_{ab} as similarity of topic *a*, tpc_a, and topic *b*, tpc_b [3]. In the first iteration, tpc_a and tpc_b contain single topic which is cluster seed taken from the extracted topics by Topics Detecting Module. For next iterations tpc_a or tpc_b could become topics cluster as a result of merge topics. This modification handles issue of articles that very likely have blended topics.

$$sim(d_i, tpc_t) = \sum_{k \in t} \frac{tf_{ik} \times (\log(\frac{N}{n_k}))^2}{\sqrt{\sum_{j \in t} (\log(\frac{N}{n_k}))^2} \times \sqrt{\sum_{j \in t} (tf_{ij})^2 (\log(\frac{N}{n_k}))^2}} \quad \text{.....(6)}$$

$$sim_{ab} = \frac{\frac{\sum_{i \in docs} sim(d_i, tpc_a) \times sim(d_i, tpc_b)}{N}}{\frac{\sum_{j \in docs} sim(d_j, tpc_a)}{N} \times \frac{\sum_{k \in docs} sim(d_k, tpc_b)}{N}} \quad \text{.....(7)}$$

Distances between two clusters of topics signify overlapping domain of subjects. Similarity function which is used to define likeness of document and cluster also works for calculating distances between clusters [3]. Our implementation demonstrates that similarity function of document and cluster is not only for merging topics [9] [11]. Aforesaid design for cutting-off topics-subtopics tree utilizes similarity function to work out on variance ratio value. In the end cluster of single topic or merged topics will have a document representation. For that purpose, this module searches the representation from list of documents similar to clusters of topics. The list is a descended order by depth-first traversing. Similarity function is used to measure likeness level similarity between document and cluster of topics based on common words which exist within. We compute similarity of each traversed document ignoring any document which has already become representation of other clusters. Similarity between document and sub clusters of currently visited cluster is also checked, because it is possible that the document is more suitable as representation of one of its sub cluster. In order to get orders of traversal topics in the list, we consider implicit links information of Web pages. PageRank [16] is selected to do link analysis for acquiring importance weight of each document representation.

IV. Adaptive Content-based Navigation Generating System

Content-based navigation generating system changes point focus of learning path if current knowledge state of users is shifting. Query Searching Module will pin-point some documents from the collection where their contents relevant with information in user model (see Fig.1). Then the system

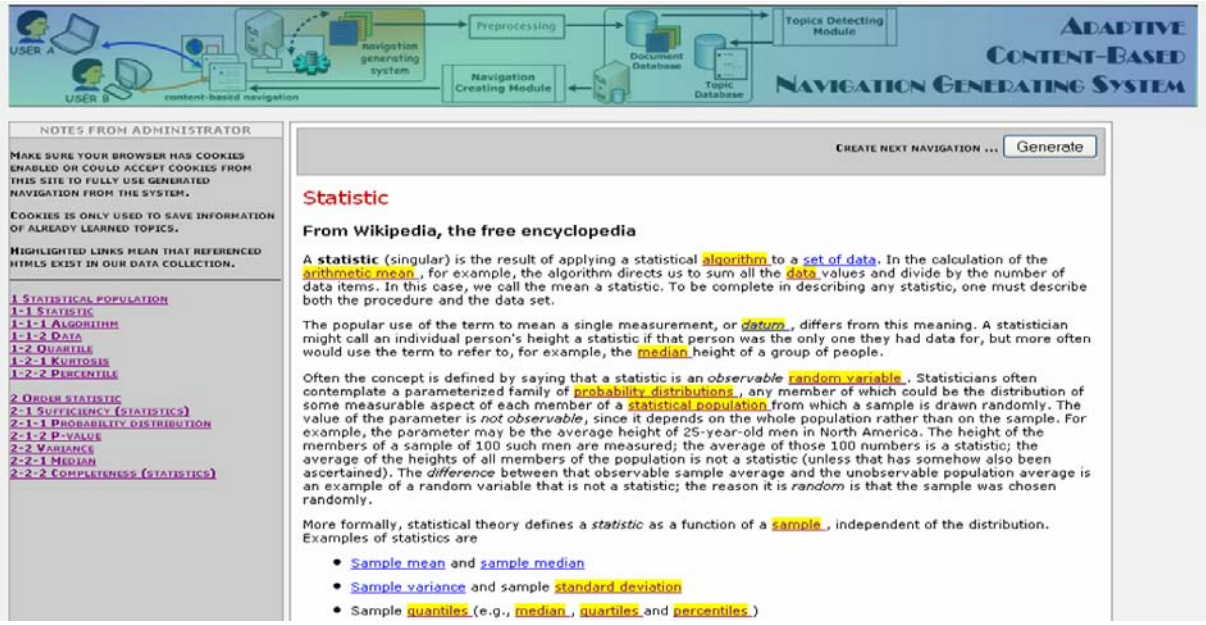

Fig. 2. Browser displays first page from a generated navigation.

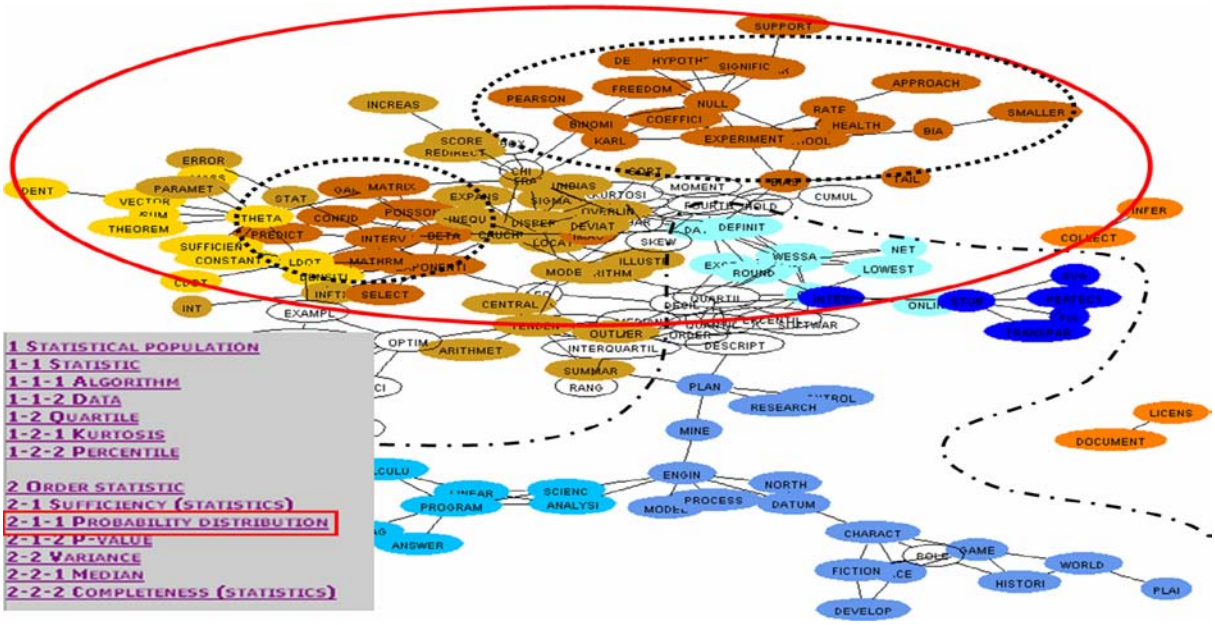

Fig. 3. Map of topics from a generated navigation on a Web-based Application.

adaptively re-generates a new navigation to learn those documents.

A. *Query Searching Module*

We use PageRank [16] to pick out most *linked-by* Web pages which their contents are more suitable with selected topic. Generated navigation is derived from mapping of clustered topics where each single topic might be term phrases. Since position of combined terms might lead to different meaning, to find suitable contents we need to exploit schema of term weighting that considers position of terms [17]. Our searching process applies combined ranking factors of link analysis, PageRank Scoring, and content analysis that still retaining *spatial* information (position of terms) of search keywords,

Fourier Domain Scoring [2]. Searching results act as dummy collection replacing the original one in which the system will produce the next suitable navigation for more focused collection to learn further.

Adaptive content-based navigation generating system uses Vector Space Model (VSM) or TF-IDF weighting, Fourier Domain Scoring (FDS) and PageRanks Score (PRS) in a combination to get similarity score of a document with selected material. The selected one is a Web page where user clicks `[Generate]` button (see Fig.2). System concerns term weight values of important terms that exist in document and query Web pages to get VSM score. System will calculate FDS score if and only if title of selected material consists more than single term. Finally multiplication between those three kinds of scores, if FDS score exists, defines final similarity between Web pages of document and query.

B. *System Implementation on Web-based Application*

As experiments to test functionality of each module, adaptive content-based navigation generating system is implemented. Small collection of documents contains English Wikipedia articles without images [1] is prepared. Collection of Web pages is prepared beforehand such as do html scraping based on XML files as patterns to extract only learning contents from raw Web pages. For this time being it is easier to observe adaptive effect with small experimental collection ($\pm$30 docs). The adaptive is concerned with topics mapping to recognize domain of subjects and then topics spotting of user interests to provide scenarios for learning subjects. We use libraries of Oracle Text [2] in indexing terms and Porter Stemmer algorithm in stemming processes. List of stop words contains words provided by Oracle Text, HTML tags and some common words in Wikipedia articles.

Browser display Web page in Fig.2 for users. Highlighted links indicate that their being referenced Web pages are documents in the collection. System deals with contents of those pages in consecutive way of routine processes from aforementioned modules to derive map of topics for generating navigation.

First generated navigation is illustrated in Fig.3 where it also shows graph of words [3], though to create illustration graph is not part of navigation generating system. Note that the graph describes map of topics in experimental collection of Wikipedia articles. The navigation has two sections as shown with a two-divided area in map op topics by dashed line. Each divided area in the graph contains several groups of gradational vertices. Nodes with the same gradation color are common words of single topic or sub section in generated navigation.

Let say user clicks one topic of first generated navigation in Fig.3 (see rectangle area of a sub section in the navigation). Selected topic has words in dotted ellipses within its coverage concept of subjects. Query Searching Module recognizes that searching results of documents have subjects relate to selected topic as illustrated with ellipse area of solid line. For second generated navigation in this case, the system will produce a structure from pin-points topics within solid line ellipse, and not from a whole map of topics. It is said that the system adaptively changes point focus of learning path for users based on their current interest.

Fig.2 illustrates Statistic course in a sample of web-based learning. It is suggested that user learns topics about [Statistic Population] then followed by [Order Statistic] from the course. If user clicks link of [Probability Distribution], the system analyzes topic mapping in Fig.3 and generates new learning path focused on topics in ellipse area with solid line. The second generated path does not concern with all topics in the mapping.

V. CONCLUSIONS AND FUTURE WORKS

In previous works we do study literatures and propose a framework of navigation generating system, evaluate selected methods in the frameworks to confirm their validities, then implement the framework with still one left aspect to produce appropriate learning path for users: adaptive effect. This work brings to completion of proposed framework for generating navigation solely based on contents in the collection of documents. With addition of adaptive effect, the system could generate navigation as guidance for users without neglecting their context of interests on certain learning subjects.

Adaptive content-based navigation generating system has produced a structural list of topics, not only hierarchical list in the different level but also reading sequence in the same level. Instead following links within Web pages of learning materials and then reading them in a hypertextual manner, users could follow path in our generated navigation and stay away of any hypertext disorientation.

System implementation on this paper utilizes the term of adaptive from user model with information on the selected topic. As a note, it is believed that the collection of learning materials consists of topics and relation between topics decided by frequency of words in documents within the collection. Information of selected topics by user should be recorded to reflect user preferences of learning topics. Preference of a user is not adequately reflected yet in this paper. More elaborate studies are needed to represent user model. Information about previously visited links, time to access in each link or others could become reference for modeling users. Next, we would like the system also considers those aspects.

The framework that has been discussed so far involves analyzing hidden topics that we believe will affect the optimal path of navigation support system to the user. It also will affect the reaction of the user to the system about learning process. Though evaluating the framework about functionality has been

[1] crawled from main page of category Statistic
http://en.wikipedia.org/wiki/Statistic
[2] Oracle Text in Oracle Database
http://www.oracle.com/technology/products/text
[3] hypergraph is written with Dot Language and viewed at ZGRViewer
http://zvtm.sourceforge.net/zgrviewer.html

done, more quantitative evaluations of the effectiveness of the system are also expected. The effectiveness should face the question of how to directly evaluate user reaction to navigation generating system. Our next work will also consider this unanswered question

REFERENCES

[1] P. Brusilovsky and E. Millan, "The Adaptive Web: Methods and Strategies of Web Personalization", ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2007, vol. 4321, ch. User Models for Adaptive Hypermedia and Adaptive Educational Systems, pp. 3–53.

[2] D. Purwitasari, Y. Okazaki, and K. Watanabe, "A Study on Web Resources' Navigation for e-Learning: Usage of Fourier Domain Scoring on Web Pages Ranking Method," in ICICIC '07: Proc. of the Second Intl. Conf. on Innovative Computing, Information and Control, 2007.

[3] C. Clifton, R. Cooley, and J. Rennie, "TopCat: Data Mining for Topic Identification in a Text Corpus," IEEE Trans. on Knowledge and Data Engineering, vol. 16, no. 8, pp. 949–964, 2004.

[4] M. E. S. Mendes, W. Jarrett, O. Prnjat, and L. Sacks, "Flexible Searching and Browsing for Telecoms Learning Material," in IST'2003: Proc. of the 2003 Intl. Symp. on Telecomm., 2003.

[5] M. Halkidi, B. Nguyen, I. Varlamis, and M. Vazirgiannis, "THESUS: Organizing Web Document Collections based on Link Semantics," The VLDB Journal, vol. 12, no. 4, pp. 320–332, 2003.

[6] J. Zhu, J. Hong, and J. G. Hughes, "PageCluster: Mining Conceptual Link Hierarchies from Web Log Files for Adaptive Web site Navigation," ACM Trans. Interet Technol., vol. 4, no. 2, pp. 185–208, 2004.

[7] A. D. Wissner-Gross, "Preparation of Topical Reading Lists from the Link Structure of Wikipedia," in ICALT '06: Proc. of the Sixth IEEE Intl. Conf. on Advanced Learning Technologies, 2006, pp. 825–829.

[8] S. Reinhold, "WikiTrails: Augmenting Wiki Structure for Collaborative, Interdisciplinary Learning," in WikiSym '06: Proc. of the 2006 Intl. Symp. on Wikis, 2006, pp. 47–58.

[9] D. Purwitasari, Y. Okazaki, and K. Watanabe, "Content-based Navigation in Web-based Learning Applications," in ICCE '08: Proc. of the 16th Intl. Conf. on Computers in Education, 2008, pp. 557–564.

[10] D. Purwitasari, Y. Okazaki, and K. Watanabe, "A Study on Adaptive Content-based Navigation Generating System for Web-based Learning," in SIG-ALST'53: Proc. of the 53 Conf. on Special Interest Group - Adv. Learning Science and Tech., The Japanese Society for Artificial Intelligence. IEEE Education Japan Chap., 2008, pp. 31–36.

[11] D. Purwitasari, Y. Okazaki, and K. Watanabe, "Data Mining for Navigation Generating System with Unorganized Web Resources," in KES '08: Proc. of the 12th Intl. Conf. on Knowledge-Based Intelligent Information and Engineering Systems, 2008, pp. 598–605.

[12] G. Karypis, "Multilevel Hypergraph Partitioning," Comput. Sci. and Eng. Dept., Univ. Minnesota, Minneapolis, Tech. Rep. 02-25, 2002.

[13] R. B. Yates and B. Ribeiro-Neto, Modern Information Retrieval. Addison-Wesley, 1999.

[14] J. R. Quinlan, "Induction of Decision Trees," Machine Learning, vol. 1, pp. 81–106, 1986.

[15] R. Agrawal, T. Imieli´nski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," ACM SIGMOD, vol. 22, no. 2, pp. 207–216, 1993.

[16] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Library Technologies Project, Tech. Rep., 1998.

[17] L. A. Park, K. Ramamohanarao, and M. Palaniswami, "Fourier Domain Scoring: A Novel Document Ranking Method," IEEE Trans. on Knowledge and Data Engineering, vol. 16, no. 5, pp. 529–539, May 2004.

.