

# FOCI: A Personalized Web Intelligence System

Ah-Hwee Tan, Hwee-Leng Ong, Hong Pan, Jamie Ng, Qiu-Xiang Li  
Kent Ridge Digital Labs  
21 Heng Mui Keng Terrace, Singapore 119613,  
email: {ahhwee, hweeleng, panhong, jamie, qiuxiang}@krdl.org.sg

## Abstract

This paper introduces a system known as Flexible Organizer for Competitive Intelligence (FOCI) that provides an integrated platform for gathering, organizing, tracking, and dissemination of competitive information on the web. FOCI enables a user to build information portfolios by gathering and organizing on-line information according to his/her needs and preferences. Through a novel method called user-configurable clustering, a user can personalize his/her portfolios in terms of the content and the information structure. The personalized portfolios can be constantly updated by tracking relevant information and new information can be organized into appropriate folders of the portfolios automatically. The personalized portfolios thus function as "living reports" that can then be published and shared by other users.

## 1 Introduction

Web intelligence can be defined as the process of scanning and tracking information on the world wide web so as to gain competitive advantages. In the knowledge-based era, it has become increasingly risky to do business without intelligence. With the popularization of World Wide Web, one can obtain a tremendous amount of information from online sources readily. However, it is still very labor intensive to compile and organize such information into actionable reports.

Popular internet search engines, such as Yahoo!, Excite, AltaVista, and Lycos, retrieve documents upon users' search queries but do not organize the search results. More sophisticated tools such as Copernics, BullsEye, and NorthernLight organize search results into automatically generated folders to facilitate navigation and browsing. However, as in typical clustering systems [Carpenter and Grossberg, 1987; Kohonen, 1988; Kaski *et al.*, 1996], users have very little control on how the information are organized and the information clusters generated may not match with the users' requirement. As a consequence, internet search tools are used

mainly for gathering purpose only. Serious users, such as intelligence scouts, still have to manually compile the materials according to their needs and preferences. This can be a painstaking process, especially when the information needs to be updated frequently.

This paper introduces a system called FOCI (for Flexible Organizer for Competitive Intelligence) to assist knowledge workers to perform competitive intelligence on the web. FOCI bridges the gap between raw search results and organized competitive information by providing an integrated platform that supports the key activities in a competitive intelligence cycle.

FOCI constructs information portfolios by gathering and organizing on-line information into automatically generated folders. A user can then annotate and personalize the portfolios in terms of the content and how the content is organized (i.e. the information structure) according to his/her needs and preferences. In FOCI, personalization is achieved through a method that incorporates users' preferences in an information clustering system [Tan and Pan, ]. The personalized portfolios can be constantly updated by tracking and organizing new information automatically. The portfolios thus function as "living reports" that can be published and shared by other users. In all, the system provides an environment for gathering, organizing, tracking, and publishing of competitive information on the web.

The rest of this article is organized as follows. Section 2 presents the FOCI architecture and a brief description of it's main components. Section 3 presents user-configurable clustering, a key enabling technology of FOCI. Section 4 illustrates how FOCI can be used in creating, organizing, and tracking an exemplary information portfolio. The final section concludes and highlights future works.

## 2 FOCI System Architecture

Referring to Figure 1, FOCI comprises an information gathering module for retrieving and integrating online information from diversified sources, a content management module for organizing and personalizing portfolios, a content mining module for analyzing information portfolios, a content publishing module for sharing of

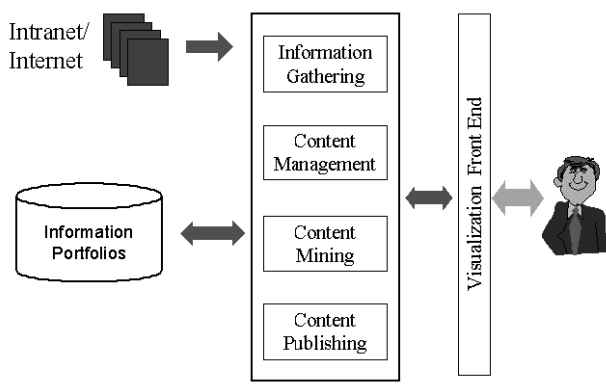


Figure 1: The FOCI system architecture.

portfolios, and a user interface module for graphical visualization and user interactions. We briefly describe the key functions supporting a competitive intelligence cycle below.

1. **Information gathering:** The information *gathering* module allows users to build an information portfolio by searching and integrating information given by major internet search engines and news sites. Users can also insert their own links or documents not found in the search results into portfolios directly. An automatic *tracking* function monitors a selected set of online sources and updates the portfolio with new content periodically.
2. **Content management:** The content management module provides a host of utilities for a user to organize and manage his/her competitive information in the preferred manner. Domain-specific template is also provided for organizing information into predefined section. In the Information Technology domain, for example, a recommended template organizes competitive information into *news*, *market*, *company*, *resources*, and *events*. Coupled with an automatic clustering engine, FOCI allows a set of personalization functions such as labeling, adding, deleting, grouping, and splitting of clusters. Additional functions include annotation and deletion of documents and portfolios.
3. **Content mining:** The content mining module extracts key attributes from raw information content and transforms them into intuitive format for information discovery. Key analysis functions include trend analysis, topic detection/tracking [Kanagasa and Tan, 2001], and link association. Due to the space constraint, content mining is outside the scope of this paper.
4. **Content publishing:** The content publishing module handles the permission control of individual information portfolios so that a user can elect to release his/her portfolios for public access. Various views are also supported for presenting portfolios in different levels of details.

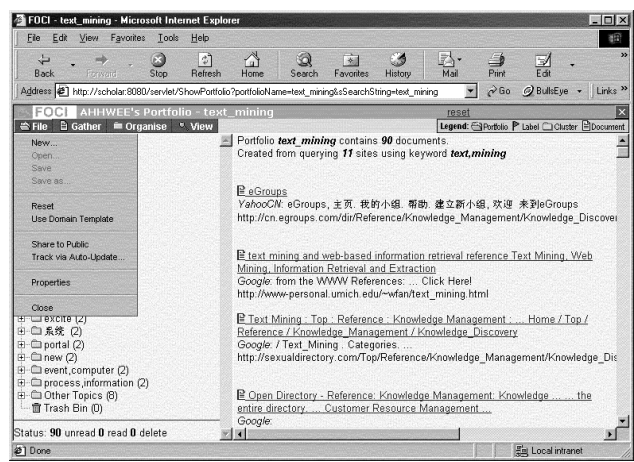


Figure 2: A screen shot of FOCI.

The FOCI server is running on UNIX SOLARIS workstations. A pre-alpha version of the system is available at <http://textmining.krdl.org.sg/FOCI>. As the user interface is based on Servlets and dynamic HTML, the application is accessible through Internet Explorer (IE) version 5.0 and above only. Figure 2 provides a screen shot of FOCI in action.

### 3 User-configurable Clustering

A user-configurable clustering system (Figure 3) comprises an information clustering engine for clustering of information based on similarities, a user interface module for displaying the information groupings and obtaining user preferences, a personalization module for defining, labeling, and modifying cluster structure, and a knowledge base for storing user-defined cluster structures.

The personalization module works in conjunction with the information clustering engine to incorporate user preferences to modify the automatically generated cluster structure. Through the user interface module and the personalization module, a user is able to perform a wide range of cluster manipulation functions including labeling existing clusters, inserting new clusters, and organizing clusters by merging and splitting clusters through the use of labels or themes. The customized cluster structure can be stored in the cluster structure knowledge base and retrieved at a later stage for processing new information. Based on the personalized cluster structure, new information can be organized accordingly to the user's preferences captured over the previous sessions.

#### 3.1 Feature Representation

To perform real-time content aggregation and clustering, we estimate the content of the pages based on the information provided on the search results pages by the search engines (instead of loading the original documents). In addition to keywords contained in the titles and descriptions of links, we also make use of the URL addresses which provide much meta information of the web pages.

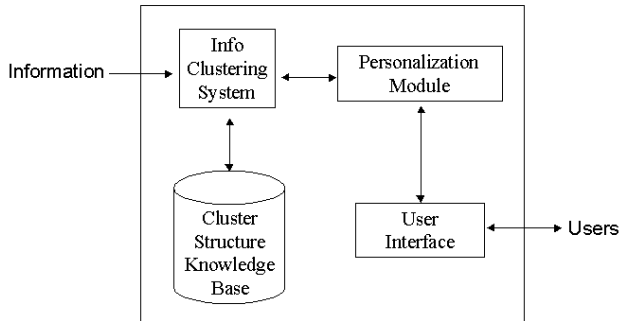


Figure 3: A user-configurable clustering system.

For each document  $d$ , we derive an information vector  $\mathbf{A} = (a_1, a_2, \dots, a_M)$  such that

$$a_i = (1 + tf(w_i)) * idf(w_i) \quad (1)$$

where the term frequency  $tf(w_i)$  is the number of times the keyword  $w_i$  appears in document  $d$  and the inverse document frequency  $idf(w_i)$  is computed by

$$idf(w_i) = \log \frac{N}{df(w_i)} \quad (2)$$

where  $N$  is the number of the documents in the collection and the document frequency  $df(w_i)$  denotes the number of documents that  $w_i$  appears in.

User preferences are represented by preference vectors that indicate the preferred groupings of the information. A preference vector  $\mathbf{B}$  is defined by

$$\mathbf{B} = (b_1, b_2, \dots, b_N) \quad (3)$$

where  $b_i$  is either zero or one, indicating the presence or absence of the user-defined label  $L_i$ .

### 3.2 Clustering Engine

The information clustering engine is based on fuzzy ARAM [Tan, 1995; Tan and Soon, 1996] that performs a combination of unsupervised learning and supervised learning. ARAM belongs to a family of predictive self-organizing neural networks known as predictive Adaptive Resonance Theory (ART) that performs incremental supervised learning of recognition categories (pattern classes) and multidimensional maps of patterns. An ARAM system can be visualized as two overlapping Adaptive Resonance Theory (ART) [Carpenter and Grossberg, 1987] modules consisting of two input fields  $F_1^a$  and  $F_1^b$  with an  $F_2$  category field (Figure 4). The ART modules used in ARAM can be ART 1 [Carpenter and Grossberg, 1987], which categorizes binary patterns, or analog ART modules such as ART 2, ART 2-A, and fuzzy ART [Carpenter *et al.*, 1991] which categorize both binary and analog patterns. Fuzzy ARAM [Tan, 1995; 2001] that is based on fuzzy ART is used in this paper.

For user-configurable clustering, the  $F_1^a$  field contains the activities of the information vectors and the  $F_1^b$  field

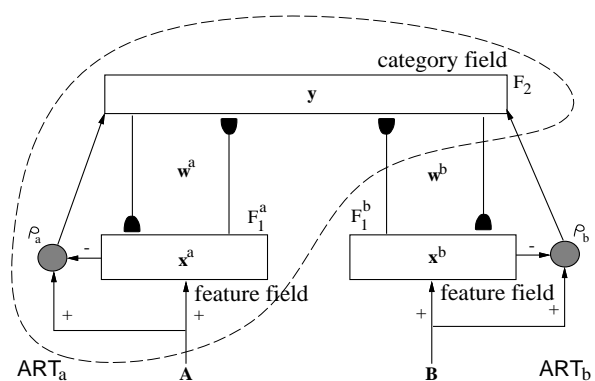


Figure 4: The Adaptive Resonance Associative Map architecture.

contains the activities of the preference vectors. Information clusters (represented at  $F_2$ ) are created during *learning* through the synchronized clustering of the information and preference vectors. Specifically, each recognition node or cluster  $j$  learns to encode a pair of template information vector  $\mathbf{w}_j^a$  and template preference vector  $\mathbf{w}_j^b$ .

### 3.3 Clustering

If a predefined cluster structure exists, the system loads the ARAM network before clustering. Otherwise, a new network is created which contains zero cluster. During clustering, for each document  $d$ , a pair of vectors  $(\mathbf{A}, \mathbf{E})$  is presented to the system, where  $\mathbf{A}$  is the information vector of  $d$  and  $\mathbf{E}$  is a null vector such that  $E_i = 0$  for  $i = 1, \dots, N$ .

Given an information vector  $\mathbf{A}$ , the system first searches for a  $F_2$  cluster  $J$  encoding a template information vector  $\mathbf{w}_j^a$  that is closest to the information vector  $\mathbf{A}$  according to a choice function. It then checks if the associated  $F_2$  template information vector  $\mathbf{w}_j^a$  of the selected category matches with the information vector according to a *match* criterion. If so, the template information vector of the  $F_2$  cluster  $J$  is modified to encode the input information. Otherwise, the cluster is reset and the system repeats to select another cluster until a match is found or a new cluster is created.

With a predefined cluster structure, fuzzy ARAM organizes the information according to the cluster structure. Without predefined network structure, fuzzy ARAM reduces to a pure clustering system that self-organizes the information based on the similarities among the information vectors only. The coarseness of the information groupings is controlled by the  $ART_a$  vigilance parameter ( $\rho_a$ ) used in the match criterion.

### 3.4 Personalization

ARAM can also operate in an *insertion* mode whereby a pair of information and preference vectors can be inserted directly into an ARAM network. Whereas *learning* mode is used for clustering and obtaining the cluster

assignments of information vectors, *insertion* mode enables a computer user to influence the clusters created by ARAM through indicating his/her own preferences in the forms of preference vectors. We present the key cluster personalization functions below.

### Labeling Information Clusters

Associating clusters with labels or themes allows a user to "mark" specific information groupings that are of interests to the user so that the information can be found readily in the future and new information can be organized according to such information groupings. Through the user interface module, a user can assign a label  $L$  to a cluster  $j$  by modifying the template preference vector  $\mathbf{w}_j^b$  to equal  $\mathbf{B}$ , where  $\mathbf{B}$  is a preference vector representing  $L$ . Labels reflect the user's interpretation of the groupings. They are useful landmarks to the user in navigating and locating old as well as new information.

### Inserting Information Clusters

A user can define and insert his/her own information groupings or clusters into an ARAM network so that the information can be organized according to such information groupings. The inserted clusters reflect the user's preferred way of grouping information and are used as the default slots for organizing information.

To insert a new cluster, a pair of information and preference vectors ( $\mathbf{A}, \mathbf{B}$ ) are first derived based on the key attributes of the information in the new cluster and the cluster label. During cluster insertion, fuzzy ARAM's vigilance parameters  $\rho_a$  and  $\rho_b$  are each set to 1 to ensure that only identical attribute vectors are grouped into one recognition category. After insertion, ARAM re-generates the cluster structures by clustering all the information vectors again. Note that new clusters may be generated during reclustering.

### Merging Information Clusters.

A merge cluster function allows a user to combine two or more information groupings generated by the clustering process. Intuitively, a user could simply refer to any number of clusters and demand them to be grouped under a cluster label or theme. To merge the clusters, the algorithm simply modifies their template preference vectors to encode a common label or theme.

### Splitting Information Clusters

A split cluster function allows a user to split an information group into two clusters in an intuitive manner. Specifically, a user can refer to two items  $d_1$  and  $d_2$  in a cluster and assign them with different labels.

To split a cluster, two pairs of information and preference vectors, namely  $(\mathbf{A}_1, \mathbf{B}_1)$  and  $(\mathbf{A}_2, \mathbf{B}_2)$  are inserted into the ARAM network, where  $\mathbf{A}_1$  and  $\mathbf{A}_2$  are the information vectors of  $d_1$  and  $d_2$  respectively, and  $\mathbf{B}_1$  and  $\mathbf{B}_2$  are preference vectors derived from the cluster labels  $L_1$  and  $L_2$  respectively. As  $\mathbf{B}_1 \neq \mathbf{B}_2$ , and  $\rho_b = 1$ ,  $\mathbf{A}_1$  and  $\mathbf{A}_2$  will be grouped accordingly into different clusters. In addition, those information vectors, originally in the same cluster, will be re-organized into one of the



Figure 5: Clusters created by fuzzy ARAM based on the 71 documents collected through the four internet search engines.

two new clusters depending on their similarities to  $\mathbf{A}_1$  and  $\mathbf{A}_2$ .

## 4 Experiments

In this section, we illustrate how FOCI can be used to create, organize, and track specific topics of interests. The objective of the experiments is twofold. First, we show how the personalization functions can be used to support a variety of organization functions. In addition, we demonstrate how a personalized portfolio can serve as a template for organizing new information.

### 4.1 Clustering

An information portfolio on "text mining" is created by integrating search results of four internet search engines. There are a total of 71 hits after removing duplicated links. Figure 5 depicts the clustering results based on a combination of URL and content-based keyword features. There are 17 clusters, each characterized by one to three keywords listed in decreasing order of importance. Three clusters, namely *fortune*, *data*, and *information*<sup>1</sup> are the most prominent ones with 20, 17, and 7 documents respectively.

### 4.2 Personalization

Based on the raw cluster structure generated, this section illustrates how a user may use the various cluster manipulation functions, namely labeling, inserting, merging, and splitting, to personalize his/her portfolios.

Figure 6 shows a partially personalized portfolio. The *fortune* cluster containing news articles from the Fortune news site has been labeled under the theme of *Fortune*

<sup>1</sup>For convenience, we refer to a cluster by the first keyword in its keyword list.

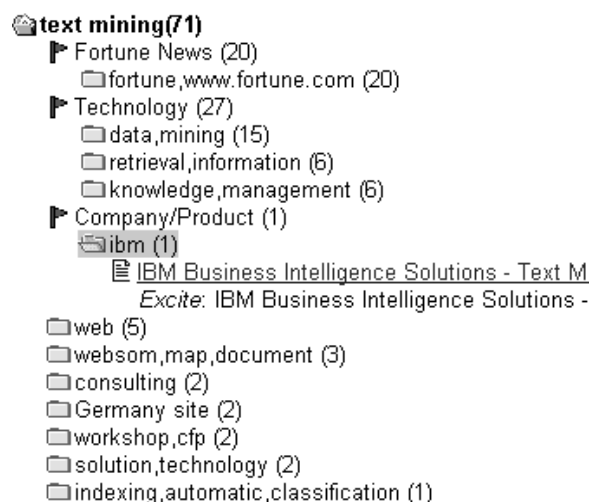


Figure 6: A partially personalized portfolio on text mining.

*News*. In addition, a number of user-defined clusters have been created under the theme of *Technology*. The documents in these user-defined clusters are mainly from the original *data*, *information*, and *knowledge* clusters in figure 5. In addition, a user-defined cluster with a keyword *IBM* under *Company/Product* manages to pull out a link to a IBM Business Intelligence/Text Mining page which was buried somewhere previously. With these user-defined clusters, new clusters have emerged. The most interesting cluster discovered is the *websom* cluster containing three links to websom related information.

Figure 7 shows an exemplary fully personalized portfolio. A number of split and merge clusters operation have been performed to organize the clusters into five themes. This portfolio has used a combination of organizing schemes. While much of the information is grouped according to their sources and the nature of the content (such as News, Company/Products, Research, and Events), there is a horizontal grouping on *Technology* that organizes information according to the various subfields and related topics in text mining, such as knowledge management, data mining, and information retrieval.

### 4.3 Tracking

In this section, we study the effect of tracking and clustering new information using the personalized portfolio. A new set of 42 documents was collected through three additional search engines. Without prior structure, the documents would be organized into the clusters as shown in figure 8. In contrast, figure 9 shows the clustering result when the new documents are organized based on the personalized cluster structure. There are 113 documents in the combined portfolio. A significant portion of the new information, especially those in the *search*, *software*, *information*, and *knowledge* clusters (figure 8), have been organized into clusters under the themes of *technology*

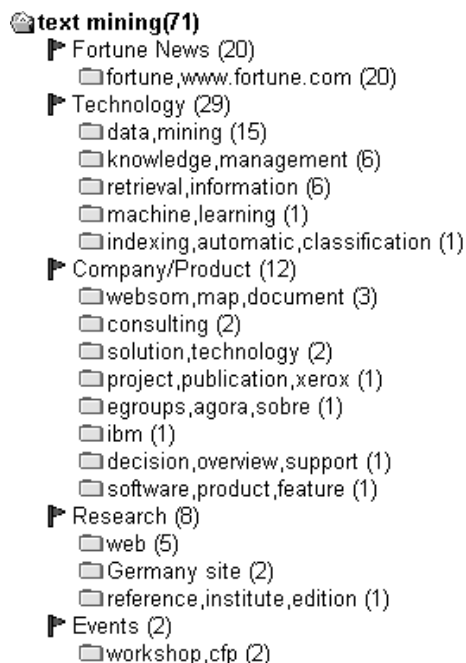


Figure 7: A personalized portfolio on text mining. All information have been organized into one of the five themes.



Figure 8: Clusters created by fuzzy ARAM based on the 42 new documents without personalization.

and *Company/Product*. Some of the other clusters remain highlighting information that do not fit into the personalized structure. The most prominent group is the *businesswire* cluster that contains news articles from the BusinessWire news site. This indicates that the system is able to discover novel information groupings while organizing familiar information into the user's personalized portfolio.

## 5 Conclusions

This paper has presented a web-based intelligence system known as FOCI that enables a user to create and manage personal information portfolios. Personalization in FOCI is achieved via user-configurable clustering that integrates the complementary strengths of clustering and categorization. The method is more flexible than a pure categorization system in which information has to be assigned to one or more pre-defined categories or groups. On the other hand, it is more manageable than a pure

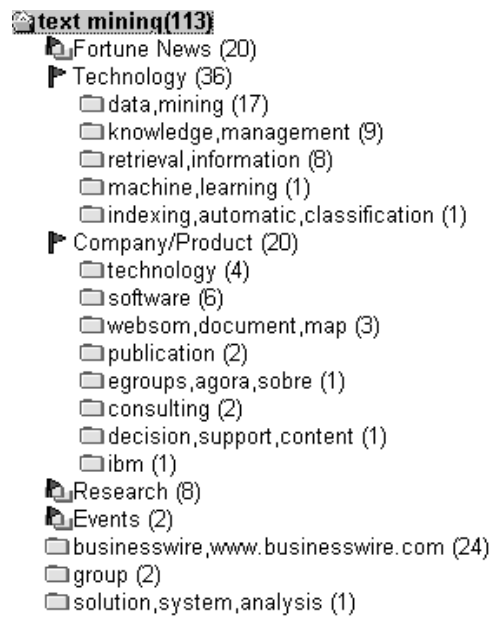


Figure 9: Organization of the new documents into the personalized portfolio.

clustering or self-organizing system in which users have very little control over how the information is organized.

One aspect that we have briefly explored is the discovery of new information or knowledge. A user defines his or her know-how and interpretation of the environment in terms of how he/she wants the information to be organized. Any information that falls outside of the defined cluster structure is thus new and potentially interesting. This helps a user to identify information that is novel with respect to his/her experience.

Our current implementation makes use of keyword-based representation. Our next version to be released soon will be based on *terms*, which may contain two or three tokens. We will also launch a Topic Detection and Tracking component that will highlight new and hot topics from a stream of online news articles.

## References

- [Carpenter and Grossberg, 1987] G. A. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37:54–115, 1987.
- [Carpenter *et al.*, 1991] G. A. Carpenter, S. Grossberg, and D. B. Rosen. Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, 4:759–771, 1991.
- [Kanagasa and Tan, 2001] R. Kanagasa and A-H. Tan. Topic detection, tracking and trend analysis using self-organizing neural networks. In *Proceedings, Fifth Pacific-Asia Conference on Knowledge Discovery and*

- Data Mining (PAKDD'01), Hong Kong*, pages 102–107, 2001.
- [Kaski *et al.*, 1996] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Creating an order in digital libraries with self-organizing maps. In *Proceedings, WCNN'96, San Diego*, 1996.
- [Kohonen, 1988] T. Kohonen. *Self-organization and Associative Memory*. Springer-Verlag, 1988.
- [Tan and Pan, ] A-H. Tan and H. Pan. Adding personality to information clustering. Submitted for publication.
- [Tan and Soon, 1996] A.-H. Tan and H.-S. Soon. Concept hierarchy memory model: A neural architecture for conceptual knowledge representation, learning, and commonsense reasoning. *International Journal of Neural Systems*, 7(3):305–319, 1996.
- [Tan, 1995] A.-H. Tan. Adaptive Resonance Associative Map. *Neural Networks*, 8(3):437–446, 1995.
- [Tan, 2001] A-H. Tan. Predictive self-organizing networks for text categorization. In *Proceedings, Fifth Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'01), Hong Kong*, pages 66–77, 2001.