Chapter II

Clustering Web Information Sources

Athena Vakali, Aristotle University of Thessaloniki, Greece

George Pallis, Aristotle University of Thessaloniki, Greece

Lefteris Angelis, Aristotle University of Thessaloniki, Greece

Abstract

The explosive growth of the Web has drastically increased information circulation and dissemination rates. As the numbers of both Web users and Web sources grow significantly every day, crucial data management issues, such as clustering on the Web, should be addressed and analyzed. Clustering has been proposed toward improving both information availability and the Web users' personalization. Clusters on the Web are either users' sessions or Web information sources, which are managed in a variation of applications and implementation test beds. This chapter focuses on the topic of clustering information over the Web in an effort to provide an overview and survey on the theoretical background and the adopted practices of the most popular emerging and challenging clustering research efforts. An up-to-date survey of the existing clustering schemes is given to be of use for both researchers and practitioners interested in the area of Web data mining.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Introduction

The explosive growth of the Web has dramatically changed the way in which information is managed and accessed. Thus, several data management solutions such as clustering have been proposed. Specifically, clustering is the process of collecting Web sources into groups so that similar objects are in the same group and dissimilar objects are in different groups.

Clustering on the Web has been proposed based on the idea of identifying homogeneous groups of objects from the values of certain attributes (variables; Jain, Murty, & Flynn, 1999). In the context of the Web, many clustering approaches have been introduced for identifying Web source clusters evaluated under a wide range of parameters (such as their size, content, or complexity). A clustering scheme is considered to be efficient if it results in reliable Web data grouping within a reasonable time.

Clustering algorithms have their origins in various areas such as statistics, pattern recognition, and machine learning. An optimal clustering scheme should mainly satisfy the following criteria:

1. **Compactness:** The data within each cluster should be as close to each other as possible. A common measure of compactness is the variance, which should be minimized.
2. **Separation:** The clusters should be widely spaced. The notion of cluster distance is commonly used for indicating the measure of separation, which should be maximized.

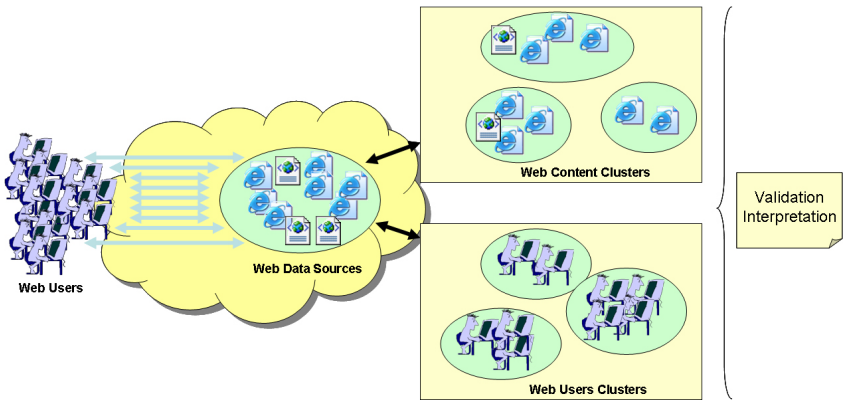
In general, the Web consists of a variety of Web sources. In order to facilitate data availability and accessing, and to meet user preferences, the Web sources are clustered with respect to a certain parameter or characteristic such as their popularity, structure, or content. Clustering on the Web can be one of the following types:

- **Web User Clustering:** The establishment of groups of users exhibiting similar browsing patterns. Such knowledge is especially useful for inferring user statistics in order to perform various actions such as market segmentation in e-commerce applications, personalized Web content for users, and so forth. This type of clustering helps in better understanding the users' navigation behavior and in improving Web users' request servicing (by decreasing the lengths in Web navigation pathways).
- **Web Document Clustering:** The grouping of documents with related content. This information is useful in various applications, for example, in Web search engines toward improving the information retrieval process (i.e., clustering Web queries). In addition, the clustering of Web documents increases Web information accessibility and improves content delivery on the Web.

Figure 1 depicts the overall clustering idea as employed on users' accessing of data over the Web. Considering the complexity and the diversity of the information sources on the Web, it is important to understand the relationships between Web data sources and Web users. Due to the fact that the Web data clustering topic is quite challenging and complex, this chapter

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Figure 1. Clustering information over the Web



contributes to understanding the role of clustering mechanisms and methodologies in accessing Web information (such as documents, users' patterns). Thus, it provides a complete view for the existing Web data clustering practices, which is essential both for computing practitioners (e.g., Web site developers) and for researchers as well.

Considerable research efforts have focused on clustering information on the Web, and earlier studies have shown that the clustering of Web sources is beneficial toward better Web data management (Baldi, Frasconi, & Smyth, 2003; Cadez, Heckerman, Meek, Smyth, & White, 2003). Some of these benefits are listed:

- **The improvement of the Web searching process:** Clustering Web content allows efficient query processing over the large amount of documents stored on Web servers.
- **The interaction with information retrieval systems:** Query clustering helps in discovering frequently asked questions or the most popular topics on a search engine.
- **The construction and maintenance of more intelligent Web servers:** Intelligent servers that are able to dynamically adapt their designs to satisfy future user needs, providing clues about improvements in site design, might be useful.
- **The improvement of caching and prefetching schemes** This will help to deliver the appropriate content (products) to the interested users in a timely, scalable, and cost-effective manner.
- **The adaptation of e-commerce sites to customers' needs:** Understanding Web users' navigation behavior through e-commerce Web sites can provide valuable insights into customer behavior, and can help end users, for example, by recommending new products to Web site visitors based on their browsing behavior.

In order to identify the Web data clusters, a number of clustering algorithms has been proposed and is available in the literature (Baldi, Franconi, & Smyth, 2003; Jain, Murty, & Flynn,

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

1999). In general terms, the existing clustering approaches do not provide an indication of the quality of their outcomes. For instance, questions such as "How many clusters are there in the data set?", "Does the resulting clustering scheme fit the data set?", and "Is there a better partitioning for the data set?" show the need for clustering result validation. However, the evaluation of the quality of a clustering algorithm is not an easy task since the correct clustering is not a priori known, and it depends on the different information sources and on the nature of the underlying applications. In this context, a validation scheme is often used for evaluating whether the objects have been assigned correctly to the resulting clusters (Stein, Eissen, & Wibrock, 2003; Zaïane, Foss, Lee, & Wang, 2002). Another aspect of cluster validation is to justify the number of clusters in a clustering result. Moreover, a further analysis of the resulting clusters is also important since it helps to extract useful information that is often hidden. For example, the experts in an application area have to integrate the clustering results with other experimental evidence and analysis in order to draw the right conclusion. Data mining techniques, statistical analysis, and visualization tools are usually used in order to interpret the clusters.

The rest of the chapter is organized as follows. The types of Web sources used for clustering are described, then a presenation of how these are processed toward clustering is given. The most representative Web data clustering schemes and algorithms are presented. An overview of the most indicative validation and interpretation techniques for clustering information over the Web is given. The most popular Web applications that are favored for clustering are highlighted. Finally, conclusions are made.

Information Sources Used for Clustering

A wide range of information sources are available on the Web. These sources might lie at the server side, at the client side, or at proxy servers. Each type of Web information collection differs not only in the location of the Web data source, but also in the formats of data available. In the following paragraphs, we classify the sources that are most commonly available on the Web and describe the way they are processed in order to be used by a clustering scheme.

Web Documents

Web documents are all the objects that are stored in Web servers around the world and can be accessed via a browser. In general, each Web site is considered as a collection of Web documents (a set of related Web resources, such as HTML [HyperText Markup Language] files, XML [eXtensible Markup Language] files, images, applets, multimedia resources, etc.). Typically, documents on the Web have a very large variety of topics; they are differently structured and most of them are not well structured. Therefore, Web documents need to be represented in an effective manner in order for them to be clustered. A typical approach is to preprocess them (either by their content or by their structure) prior to clustering.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Web Server Logs

A Web user may visit a Web site from time to time and spend an arbitrary amount of time between consecutive visits. All this traffic is logged in a Web server-side log file. In particular, a common log file of any given Web server is a simple text file with one user access record per line. Each user access record consists of the following fields: the user's IP (Internet protocol) address (or host name), the access time, the request method (e.g., GET, POST, etc.), the URL (uniform resource locator) of the document accessed, the protocol, the return code, and the number of bytes transmitted. The format of a common log-file line has the following fields separated by a space:

[remotehost rfc931 authuser date request status bytes]

- **remotehost:** The remote host name (or IP address number if the DNS [domain name system] host name is not available or was not provided);
- **rfc931:** The remote log-in name of the user (if not available, a minus sign is typically placed in the field);
- **authuser:** The user name with which the user has authenticated himself or herself (if not available, a minus sign is typically placed in the field);
- **date:** Date and time of the request;
- **request:** The request line exactly as it came from the client (i.e., the file name and the method used to retrieve it, typically GET);
- **status:** The HTTP (hypertext transfer protocol) response code returned to the client. It indicates whether or not the file was successfully retrieved, and if not, what error message was returned;
- **bytes:** The number of bytes transferred.

The access logs provide most of the data needed for Web servers' workload characterization. However, they do not provide all of the information that is of interest, such as identifying the Web users' navigation patterns, and certain processing should take place before getting valuable information from Web logs.

Web Proxy Logs

A Web proxy acts as an intermediate level of caching between client browsers and Web servers. Proxy caching can be used to reduce the loading time of a Web document experienced by users as well as the network traffic load at the server and client sides (Pallis, Vakali, Angelis, & Hacid, 2003). Proxy traces may reveal the actual HTTP requests from multiple clients to multiple Web servers. This may serve as a data source for characterizing the browsing behavior of a group of anonymous users sharing a common proxy server.

Proxy servers can be configured to record (in an access log) information about all of the requests and responses processed by the Web servers. Specifically, a proxy log file records

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

all the requests made to Web documents by a certain population of users (e.g., the set of users of a certain Internet service provider). Each line from the access log contains information on a single request for a document. From each log entry, it is possible to determine the name of the host machine making the request, the time that the request was made, and the name of the requested document. The entry also provides information about the server's response to this request, such as if the server was able to satisfy the request (if not, a reason why the response was unsuccessful is given) and the number of bytes transmitted by the server, if any. The access logs provide most of the data needed for workload characterization studies of Web servers. The format of a proxy log-file line consists of the following fields separated by a space:

[time duration remotehost code bytes method URL rfc931 peerstatus/peerhost type]

- **time:** The time when the client socket was closed. The format is Unix time (seconds since January 1, 1970) with millisecond resolution;
- **duration:** The elapsed time of the request, in milliseconds. This is the time between the acceptance and close of the client socket;
- **remotehost:** The client IP address;
- **code:** It encodes the transaction result. The cache result of the request contains information on the kind of request, how it was satisfied, or in what way it failed;
- **bytes:** The amount of data delivered to the client;
- **method:** The HTTP request method;
- **URL:** The requested URL;
- **rfc931:** The remote log-in name of the user (if not available, a minus sign is typically placed in the field);
- **peerstatus/peerhost:** A description of how and where the requested object was fetched;
- **type:** The content type of the object as seen in the HTTP reply header (if not available, a minus sign is typically placed in the field).

Information Processing Toward Clustering

Document Preprocessing

The clustering of documents depends on the quality of the representation of the documents' content. This representation is characterized by the amount and type of information to be encapsulated, and, in practice, the most important features from each document should be extracted (Moore et al., 1997). However, since each Web document has a variety of content formats (such as text, graphics, scripts), feature extraction should be facilitated by evicting

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

useless content. Thus, the so-called cleaning process is an important part of preprocessing and involves several tasks including parsing, decoding encoded characters, removing tags, and detecting word and sentence boundaries. Some learning mechanisms to recognize banner ads and redundant and irrelevant links to Web documents have already been discussed in Jushmerick (1999) and Bar-Yossef and Rajagopalan (2002), in which the preprocessing of Web documents is defined as a frequent template-detection problem (a frequency-based data mining algorithm detects templates as noise).

After cleaning, each Web document might be represented by a vector or a graph (Hammouda & Kamel, 2004; Yang & Pedersen, 1997; Zamir, Etzioni, Madanim, & Karp, 1997). The goal here is to transform each Web document (unstructured format) into a structured format using a vector of feature or attribute values (which may be binary, nominal, ordinal, interval, or ratio variables). Most document clustering methods (Baldi et al., 2003; Chakrabarti, 2003; Jain et al, 1999; Modha & Sprangler, 2003) that are in use today are based on the vector space model (VSM), which is a very widely used data model for text classification and clustering (Salton, Wong, & Yang, 1975). In particular, the VSM represents documents as feature vectors of the terms (words) that appear in all of the document sets, and each such feature vector is assigned term weights (usually term frequencies) related to the terms appearing in that document. In its simplest form, each document is represented by the (TF) vector $v_{tf} = (tf_1, tf_2, \ldots, tf_V)$, where tf_i is the frequency of the *i*th term in the document. Normally, very common words are stripped out completely and different forms of a word are reduced to one canonical form. Finally, in order to account for documents of different lengths, each document vector is usually normalized so that it is of unit length. Then, the dissimilarity between two Web documents is measured by applying a metric (such as Euclidean or Manhattan distance) or a cost function to their feature vectors.

Web Server Log Preprocessing

Web server access logs undergo a certain preprocessing, such as data cleaning and session identification. Data cleaning removes the records that do not include useful information for the users' navigation behavior, such as graphics, javascripts, small pictures of buttons, advertisements, and so forth. The remaining document requests are usually categorized into different categories.

Users' Session Identification

A user session is defined as a sequence of requests made by a single user over a certain navigation period, and a user may have a single or multiple sessions during a period of time. The most popular session identification methods include the following:

- Use a time-out threshold, in which a user poses a sequence of consecutive requests that are separated by an interval less than a predefined threshold. This session identification suffers from the difficulty of setting the time threshold since different users may have different navigation behaviors, and their time intervals between sessions may

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

significantly vary. In order to define the optimal time threshold, earlier research efforts proposed a time threshold of 25.5 minutes based on empirical data (Catledge & Pitkow, 1995), whereas Goker and He (2000) used a wide range of values and concluded that a time range of 10 to 15 minutes was an optimal session interval threshold. In general, the optimal time threshold clearly depends on the specific context and application. Up to now, the most common choice was to use 30 minutes as a default time threshold.

- Consider the reference length (Cooley, Mobasher, & Srivastava, 1999), that is, identify sessions by the amount of time a user spends on viewing that document for a specific log entry. The reference-length session identification is based on the assumption that the amount of time a user spends on a document correlates to whether the document should be classified as an auxiliary or content document for that user. In addition, in M. S. Chen, Park, and Yu (1998), the users' sessions are identified by their maximal forward reference. In this approach, each session is defined as the set of documents from the first document in a request sequence to the final document before a backward reference is made. Here, a backward reference is defined to be a document that has already occurred in the current session. One advantage of the maximal forward reference method is that it does not have any administrative parameters (e.g., time threshold). However, it has the significant drawback that backward references may not be recorded by the server if caching is enabled at the client site.
- Identify dynamically the sessions' boundaries (X. Huang, Peng, An, & Schuurmans, 2004) based on an information-theoretic approach by which session boundary detection is based on a statistical *n*-gram language modeling. In particular, this model predicts the probability of natural requests' sequences. According to this approach, a session boundary is identified by measuring the change of information (known as entropy) in the sequence of requests. Specifically, when a new object is observed in the sequence, an increase in the entropy of the sequence is observed. Therefore, such an entropy increase serves as a natural signal for session boundary detection, and if the change in entropy passes a specific threshold, a session boundary is placed before the new object.

Web Proxy Log Preprocessing

These data should also be preprocessed in order to extract useful conclusions for the workload and characterize the entire structure of the Web (Pallis et al., 2003). In general, the Web proxy logs are more difficult to manage than the Web server ones. Thus, a wide range of tools[1] has been implemented in order to manage the Web proxy log file in an efficient way. Furthermore, the Web proxy logs are preprocessed in order to extract users' sessions from them. A lot of approaches have been developed in order to identify users' sessions from Web access logs. However, these approaches may lead to poor performance in the context of proxy Web log mining. In Lou, Liu, Lu, and Yang (2002), an algorithm is proposed, called cut-and-pick, for identifying users' sessions from Web proxy logs. According to this algorithm, the sessions' boundaries are determined by using a Web site clustering algorithm based on site traversal graphs constructed from the proxy logs. In particular, if two consecutive document requests in a proxy log visit two Web sites that fall in two clusters, the two visits are regarded as irrelevant and are therefore classified into two user sessions.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Clustering Algorithms

Identifying Web Document Clusters

The main contribution of grouping Web documents is to improve both Web information retrieval (e.g., search engines) and content delivery. The clustering of Web documents helps to discover groups of documents having related content. In general, the process of grouping Web documents into categories is a usual practice (Cadez et al., 2003; Pallis, Angelis, Vakali, & Pokorny, 2004) since it improves data management and, in addition, eliminates the complexity of the underlying problem (since the number of document categories is smaller than the number of Web documents in a Web site). The approaches that have been proposed in order to group Web documents into categories can be summarized as follows (Baldi et al., 2003):

- **Content based:** The individual documents are grouped into semantically similar groups (as determined by the Web site administrator).
- **Functionality based:** Scanning for specific keywords that occur in the URL string of the document request makes the assignment of the document requests to a category.
- **Directory based:** The documents are categorized according to the directory of the Web server where they have been stored.

The schemes that have been developed for clustering Web documents can be categorized into the following two types.

Text-Based Clustering Approach

The text-based clustering approach uses textual document content to estimate the similarity among documents. In text-based clustering, the Web documents are usually represented by VSMs in a high-dimensional vector space where terms are associated with vector components. Once the Web documents are vectorized, clustering methods of vectors provide Web document clusters (Jain et al, 2003; Modha & Spangler, 2003; Wong & Fu, 2000). Similarity between documents is measured using one of several similarity measures that are based on such vectors. Examples include the cosine measure and the Jacard measure (Jain et al.). However, clustering methods based on this model make use of single-term analysis only. In Hammouda and Kamel (2004) and Zamir et al. (1997), the similarity between documents is based on matching phrases (sequences of words) rather than single words. A drawback of all these approaches is that they are time consuming since it is required to decompose the texts into terms.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Link-Based Clustering Approach

According to this approach, the Web is treated as a directed graph, where the nodes represent the Web documents with URL addresses and the edges among nodes represent the hyperlinks among Web documents. Link-based techniques use the Web site topology in order to cluster the Web documents. In Masada, Takasu, and Adachi (2004), the Web documents are grouped based only on hyperlink structure. Specifically, each cluster is considered to be a subset of a strongly connected component. In Zhu, Hong, and Hughes (2004), the authors presented a hierarchical clustering algorithm, called PageCluster, that clusters documents on each conceptual level of the link hierarchy based on the in-link and out-link similarities between these documents. The link hierarchy of each Web site is constructed by using the Web server log files.

In the same context, other works use link-based clustering techniques in order to identify Web communities (Flake, Tarjan, & Tsioutsiouliklis, 2004). A Web community is defined as a set of Web documents that link to more Web documents in the community than to documents outside of the community. A Web community enables Web crawlers to effectively focus on narrow but topically related subsets of the Web. In this framework, a lot of research has been devoted to efficiently identifying them. In Flake et al., communities can be efficiently computed by calculating the s-t minimum cut of the Web site graph (s and t denote the source and sink nodes, respectively). In Ino, Kudo, and Nakamura (2005), the authors propose a hierarchical partitioning through repeating partitioning and contraction. Finally, an efficient method for identifying a subclass of communities is given. A different technique for discovering communities from the graph structure of Web documents has been proposed in Reddy and Kitsuregawa (2001). The idea is that the set of documents composes a complete bipartite graph such that every hub document contains a link to all authorities. An algorithm for computing Web communities defined as complete bipartite graphs is also proposed. In Greco, Greco, and Zumpano (2004), the authors study the evolution of Web communities and find interesting properties. A new technique for identifying them is proposed on the basis of the above properties.

The notion of Web communities has also been used (implicitly or explicitly) in other contexts as well, but with different meanings and different objectives. For instance, there is a growing interest in compound documents and logical information units (Eiron & McCurley, 2003). A compound document is a logical document authored by (usually) one author presenting an extremely coherent body of material on a single topic, which is split across multiple nodes (URLs). A necessary condition for a set of Web documents to form a compound document is that their link graph should contain a vertex that has a path to every other part of the document. Similarly, a logical information unit is not a single Web document, but it is a connected subgraph corresponding to one logical document, organized into a set of documents connected via links provided by the document author as standard navigation routes.

Identifying XML Document Clusters

With the standardization of XML as an information exchange language over the Web,[2] documents formatted in XML have become quite popular. Similarly, clustering XML

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

documents refers to the application of clustering algorithms in order to detect groups that share similar characteristics. Although there have been considerable works on clustering Web documents, new approaches are being proposed in order to exploit the advantages that offers the XML standard. The existing approaches for clustering XML documents are classified as follows:

- **Text-based approach:** The clustering of XML documents is based on the application of traditional information-retrieval techniques (Baeza-Yates & Ribiero-Neto, 1999) in order to define distance metrics that capture the content similarity for pieces of text. Text-based approaches aim at grouping the XML documents of similar topics together. The existing approaches should consider both statistical information for the various parts of the XML documents (e.g., the frequency of a term) and hierarchical indexes for calculating efficiently the distance metrics.
- **Link-based approach:** It is based on distances that estimate similarity in terms of the structural relationships of the elements in XML documents. In this approach, each document is represented by a tree model. So, the clustering problem is replaced by a tree-clustering one. Therefore, most research works focus on finding tree edit distances in order to define metrics that capture structural similarity. Recently, in Nierman and Jagadish (2002), a method was proposed to cluster XML documents according to the structural similarity between trees using the edit distance. A quite different approach is presented in Lian, Cheung, Mamoulis, & Yiu (2004), where the XML document is represented as a structured graph (s-graph), and a distance metric is used to find similarities.

Identifying Web User Clusters

In order to cluster the Web users' sessions, each one is usually represented by an *n*-dimensional vector, where *n* is the number of Web pages in the session. The values of each vector are the requested Web pages. For simplicity, it is common to group the pages into groups. In addition, a user session may be represented by a graph where the nodes are the visited pages (Baldi et al., 2003; Lou et al., 2002). Up to now, several clustering algorithms have been proposed assigning the Web users' sessions with common characteristics into the same cluster (Jain et al, 1999). These may be classified into the following approaches:

- **Similarity-Based Approach:** In order to decide whether two sessions are clustered together, a distance function (similarity measure) must be defined in advance. Distance functions (e.g., Euclidean, Manhattan, Levenshtein [Scherbina & Kuznetsov, 2004], etc.) can be determined either directly or indirectly, although the latter is more common in applications. Hierarchical and partitional approaches are the most indicative that belong to this category. Hierarchical clustering approaches proceed successfully by either merging smaller clusters into larger ones (agglomerative methods) or splitting larger clusters (divisive methods). In general, differences among the techniques that use hierarchical clustering arise mainly because of the various ways of defining distance (similarity) between two individuals (sessions) or between two groups of individuals.[3]

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Since the distances have been computed, a hierarchical clustering algorithm is used either to merge or to divide the sessions. The result is represented by a tree of clusters (a two-dimensional diagram that is called a dendrogram) and illustrates the relations among them. On the other hand, the partitional algorithms determine a flat clustering into a specific number of clusters (e.g., k-means, k-mode, etc.). Specifically, a partition-based clustering scheme decomposes the data set into a predefined set of disjoint clusters such that the individuals within each cluster are as homogeneous as possible. Homogeneity is determined by an appropriate score function, such as the distance between each individual and the centroid of the cluster to which it is assigned.

- **Model-Based Approach:** Model-based clustering is a framework that combines cluster analysis with probabilistic techniques. The objects in such an approach are supposed to follow a finite mixture of probability distributions such that each component distribution expresses a cluster (each cluster has a data-generating model with different parameters for each cluster). The issue in model-based approaches is to learn the parameters for each cluster. Then, the objects are assigned to clusters using a hard assignment policy.[4] In order to learn the set of parameters for each cluster, the expectation-maximization (EM) algorithm is usually used. The EM algorithm originates from Dempster, Laird, and Rubin (1977). In Cadez et al. (2003), a method for employing EM on users' sessions is proposed. The EM algorithm is an iterative procedure that finds the maximum-likelihood estimates of the parameter vector by repeating the following steps:
  - The expectation E-step: Given a set of parameter estimates, the E-step calculates the conditional expectation of the complete data-log likelihood given the observed data and the parameter estimates.
  - The maximization M-step: Given a complete data-log likelihood, the M-step finds the parameter estimates to maximize the complete data-log likelihood from the E-step.

The two steps are iterated until convergence. The complexity of the EM algorithm depends on the complexity of the E- and M-steps at each iteration (Dempster et al., 1977). It is important to note that the number of clusters on model-based schemes is estimated by using probabilistic techniques. Specifically, the BIC (Bayesian information criterion) and AIC (Akaike information criterion) are widely used (Fraley & Raftery, 1998).

Similarity Based vs. Model Based

The benefits of similarity-based algorithms are their simplicity and their low complexity. However, a drawback of these algorithms is that they do not contain a metric about the structure of the data being clustered. For instance, in hierarchical approaches, the entire hierarchy should be explored a priori, and for partitioning approaches, it is essential to predetermine the appropriate number of clusters. On the other hand, the model-based approaches try to solve the above problems by building models that describe the browsing behavior of users on the Web. Modeling can generate insight into how the users use the Web as well as provide mechanisms for making predictions for a variety of applications (such as Web prefetching,

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Table 1. Web data clustering approaches

Information Source: Web Documents		
Research Work	**Cluster Content**	**Clustering Approach**
k-means (Modha & Spangler, 2003)	*Web documents*	*Text based*
Suffix-Tree Clustering (Zamir et al, 1997)	*Web documents*	*Text based*
Hierarchical Clustering Algorithm (Wong & Fu, 2000)	*Web documents*	*Text based*
Similarity Histogram-Based Clustering (SHC; Hammouda & Kamel, 2004)	*Web documents*	*Text based*
Strongly Connected Components Clustering (Masada et al., 2004)	*Web documents*	*Link based*
The s-t Minimum Cut Algorithm (Flake et al., 2004)	*Web communities*	*Link based*
PageCluster (Zhu et al., 2004)	*Web documents*	*Link based*
Distance-Based Clustering Algorithm (Baeza-Yates & Ribiero-Neto, 1999)	*XML documents*	*Text based*
S-GRACE clustering algorithm (Lian et al., 2004)	*XML documents*	*Link based*
Information Source: Web Server Logs		
Research Work	**Cluster Content**	**Clustering Approach**
Sequence-Alignment Method (SAM; Wang & Zaïane, 2002)	*Web users' sessions*	*Similarity based*
Generalization-Based Clustering (Fu, Sandhu, & Shih, 1999)	*Web users' sessions*	*Similarity based*
Weighted Longest Common Subsequences Clustering (Banerjee & Ghosh, 2001)	*Web users' sessions*	*Similarity based*
Cube-Model Clustering (Huang, Ng, Cheung, Ng, & Ching, 2001)	*Web users' sessions*	*Similarity based*
Path-Mining Clustering (Shahabi, Zarkesh, Adibi, & Shah, 1997)	*Web users' sessions*	*Similarity based*
Hierarchical Clustering Algorithm (Scherbina & Kuznetsov, 2004)	*Web users' sessions*	*Similarity based*
EM (Cadez et al., 2003)	*Web users' sessions*	*Model based*
Self-Organizing Maps (SOMs) Clustering (Smith & Ng, 2003)	*Web users' sessions*	*Model based*

the personalization of Web content, etc.). Therefore, the model-based schemes are usually favored for clustering Web users' sessions.

In fact, there are a number of reasons why probabilistic modeling is usually selected for describing the dynamic evolution of the Web instead of the other clustering approaches (Baldi et al., 2003). First of all, model-based schemes enable the compact representation of complex data sets (such as Web log files) by being able to exploit regularities present in many real-world systems and the data associated with these systems. Second, model-based

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

schemes can deal with uncertainty and unknown attributes, which is often the typical case in Web data applications. The Web is a high-dimensional system, where the measurement of all relevant variables becomes unrealistic, so most of the variables remain hidden and must be revealed using probabilistic methods. Furthermore, the probabilistic models are supported by a sound mathematical background. Another advantage is that model-based schemes can utilize prior knowledge about the domain of interest and combine this knowledge with observed data to build a complete model.

Table 1 presents a summary of the Web data clustering approaches.

Validation and Interpretation of Clusters

One of the main challenges with clustering algorithms is that it is difficult to assess the quality of the resulted clusters (Chen & Liu, 2003; Halkidi, Batistakis, & Vazirgiannis, 2002a, 2002b; Pallis et al., 2004). So, an important issue is the evaluation and validation of a clustering scheme.

Another major challenge with clustering algorithms is to efficiently interpret the resulting clusters. No matter how effective a clustering algorithm is, the clustering process might be proven to be inefficient if it is not accompanied by a sophisticated interpretation of the clusters. An analysis of the clusters can provide valuable insights about users' navigation behavior and about the Web site structure. In the following paragraphs, the most representative validating and interpreting approaches are presented.

Clustering Validation

In general, a validation approach is used to decide whether a clustering scheme is valid or not. A cluster validity framework provides insights into the outcomes of the clustering algorithms and assesses the quality of them. Furthermore, a validation technique may be used in order to determine the number of clusters in a clustering result (Fraley & Raftery, 1998).

Most of the existing validation approaches for Web data clustering rely on statistical hypothesis testing (Halkidi et al., 2002a, 2002b). The basic idea is to test whether the points of a data set are randomly structured or not. This analysis involves a null hypothesis (H_o) expressed as a statement of a random structure of a data set. To test this hypothesis, statistical tests are widely used, which lead to a computationally complex procedure. In the literature (Halkidi et al., 2002a, 2002b), several statistical tests have been proposed for clustering validation, such as Rand statistic (R), cophenetic correlation coefficient (CPCC), and the χ^2 test (Pallis et al., 2004). The major drawback of all these approaches is their high computational demands.

A different approach for evaluating cluster validity is to compare the underlying clustering algorithm with other clustering schemes, modifying only the parameter values. The challenge is to choose the best clustering scheme from a set of defined schemes according to a prespecified criterion, the so-called cluster validation index (a number indicating the quality of a given clustering). Several cluster validation indices have been proposed in the literature.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

The most indicative are the Davies-Bouldin index (DB; Günter & Bunke, 2003), Frobenius norm (Z. Huang et al., 2001), and SD validity index (Halkidi et al., 2002a, 2002b).

Clustering Interpretation

It is quite probable that the information that is obtained by the clusters needs further analysis, such as in cases involving clusters of Web users' sessions for a commercial Web site, which without any analysis may not provide useful conclusions. An interpretation of the resulting clusters could be important for a number of tasks, such as managing the Web site, identifying malicious visitors, and targeted advertising. It also helps in understanding the Web users' navigation behavior, and therefore helps in organizing the Web site to better suit the users' needs. Furthermore, interpreting the results of Web data clusters contributes to identify and provide customized services and recommendations to Web users. However, the interpretation of clusters is a difficult and time-consuming process due to large-scale data sets and their complexity.

Several research works in various industrial and academic research communities are focusing on interpreting Web data clusters (e.g., Cadez et al, 2003; Wu, Yu, & Ballman, 1998). Statistical methods are usually used in order to interpret the resulting clusters and extract valuable information. For example, a further analysis of the Web users' session clusters may reveal interesting relations among clusters and the documents that users visit (Pallis et al., 2004).

A valuable help in cluster interpretation is visualization, which can help the Web administrators to visually perceive the clustered results and sometimes discover hidden patterns in data. In Vesanto and Alhoniemi (2000), a visualization method is used in order to interpret Web document clusters based on the self-organizing map. The SOM is an artificial neural-network model that is well suited for mapping high-dimensional data into a two-dimensional representation space where clusters can be identified. However, it requires the preprocessing and normalization of the data, and the prior specification of the number of clusters. Furthermore, in Gomory, Hoch, Lee, Podlaseck, and Schonberg (1999), a parallel coordinate system has been deployed for the interpretation and analysis of users' navigation sessions of online stores. They define microconversion rates as metrics in e-commerce analysis in order to understand the effectiveness of marketing and merchandising efforts. Moreover, a tool called INSITE has also been developed for knowledge discovery from users' Web site navigation in a real-time fashion (Shahabi, Faisal, Kashani, & Faruque, 2000). INSITE visualizes the result of clustering users' navigation paths in real time. In Cadez et al. (2003), a mixture of Markov models is used to predict the behavior of user clusters and visualize the classification of users. The authors have developed a tool, called WebCANVAS (Web Clustering ANalysis and VisuAlization Sequence), that visualizes user navigation paths in each cluster. In this system, user sessions are represented using categories of general topics for Web documents. Another graphical tool, called CLUTO[5]: A clustering toolkit (software for clustering high-dimensional datasets), has been implemented for clustering data sets and for analyzing the characteristics of the various clusters. Finally, in Pallis, Angelis, and Vakali (2005) a visualization method for interpreting the clustering results is presented, revealing interesting features for Web users' navigation behavior and their interaction with

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

the content and structure of Web sites. This method is based on a statistical method, namely, the correspondence analysis (CO-AN), which is used for picturing both the intercluster and intracluster associations.

Integrating Clustering In Applications

A wide range of Web applications can be favored for clustering. Specifically, clustering schemes may be adopted in Web applications in order to manage effectively the large collections of data. Such applications include the following:

- **Web Personalization Systems:** In general, Web personalization is defined by Mobasher, Cooley, and Srivastava (2000) as any action that adapts the information or services provided by a Web site to the needs of a particular user or a set of users, taking advantage of the knowledge gained from the users' navigational behavior and individual interests in combination with the content and the structure of the Web site. The challenge of a Web personalization system is to provide users with the information they want without expecting them to ask for it explicitly. Personalization effectiveness heavily relies on user-profile reliability, which, in turn, depends on the accuracy with which user navigation behavior is modeled. In this context, the clustering of Web users' sessions improves significantly this process since an analysis of the resulting clusters helps in modeling and understanding better human behavior on the Web (Baldi et al., 2003; Cadez et al., 2003; Spiliopoulou & Faulstich, 1998).
- **Web Prefetching:** Web prefetching is the process of predicting future requests for Web objects and bringing those objects into the cache in the background before an explicit request is made for them (Nanopoulos, Katsaros, & Manolopoulos, 2003). Therefore, for a prefetching scheme to be effective, there should be an efficient method to predict users' requests. Sophisticated clustering schemes may be adopted in Web prefetching systems, reducing the user-perceived Web latency and improving the content-management process. The prefetching process is facilitated by determining clusters of Web documents that are probably requested together. In addition, clustering Web users' sessions helps in predicting the future requests so that objects can be prefetched before a request is made for them.
- **Web Search Engines:** Search engines are the most widely used tools for retrieving Web data. Their goal is to crawl over the Web and retrieve the requested documents with low communication costs in a reasonable interval of time. Recently, Web search engines have enhanced sophisticated clustering schemes in their infrastructures in order to improve the Web search process (Chakrabarti, 2003). The objects are clustered either by their popularity statistics or by their structure. Considerable work has also been done on clustering Web queries (Wen, Nie, & Zhang, 2001) and Web search results (Zeng, He, Chen, Ma, & Ma, 2004) toward the improvement of user satisfaction.
- **E-Mail Mining:** E-mail overload has grown significantly over the past years, becoming a personal headache for users and a financial issue for companies. In order to alleviate

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

this problem, Web mining practices have been developed that compute the behavior profiles or models of user e-mail accounts (Vel, Anderson, Corney, & Mohay, 2001). Thus, e-mail clustering is useful for report generation and the summarization of e-mail archives, as well as for detecting spam mail.

- **Content Delivery Networks:** Content (different types of information) delivery over the Web has become a mostly crucial practice in improving Web performance. Content delivery networks (CDNs) have been proposed to maximize bandwidth, improve accessibility, and maintain correctness through content replication. Web data clustering techniques seem to offer an effective trend for CDNs since CDNs manage large collections of data over highly distributed infrastructures (Pallis & Vakali, 2006).

Table 2 highlights some indicative Web applications and systems that have been favored for clustering in an effort to understand the importance and the challenge in adopting clustering under frameworks.

Table 2. Integrating Web data clustering on Web applications

Web Applications	Systems	Improve Information Retrieval	Reduce Traffic	Improve Quality of Service	Improve Content Management	Improve Security
Web personalization	WebPersonalizer (Mobasher et al., 2000), NETMIND[6] (a commercial system from Mindlab that produces multi-user recommendations), WUM (Web usage miner; Spiliopoulou & Faulstich, 1998), SpeedTracer (Wu et al., 1998)	√		√		√
Web prefetching	CacheFlow, NetSonic, Webcelerator		√	√	√	
Search engines	Google, Niagara[7]	√		√	√	
E-mail mining	Popfile,[8] SwiftFile, eMailSift			√		√
CDNs	Akamai,[9] Limelight Network,[10] Mirror Image[11]		√	√	√	

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Conclusion

The explosive growth of the Web has dramatically changed the way in which information is managed and accessed. Web data mining is an evolving field of high interest to a wide academic and technical community. In this framework, clustering data on the Web has become an emerging research area, raising new difficulties and challenges for the Web community. This chapter addresses the issues involved in the effect of Web data clustering on increasing Web information accessibility, decreasing lengths in navigation patterns, improving user servicing, integrating various data representation standards, and extending current Web information organization practices. Furthermore, the most popular methodologies and implementations in terms of Web data clustering are presented.

In summary, clustering is an interesting, useful, and challenging problem. Although, a great deal of research works exists, there is a lot of room for improvement in both theoretical and practical applications. For instance, the emergence of the XML standard has resulted in the development of new clustering schemes. Finally, the rich assortment of dynamic and interactive services on the Web, such as video and audio conferencing, e-commerce, and distance learning, has opened new research issues in terms of Web data clustering.

References

Baeza-Yates, R., & Ribiero-Neto, B. (1999). Modern information retrieval. Boston: Addison-Wesley.

Baldi, P., Frasconi, P., & Smyth, P. (2003). *Modeling the Internet and the Web.* New York: Wiley.

Banerjee, A., & Ghosh, J. (2001). Clickstream clustering using weighted longest common subsequences. *Proceedings of the Workshop on Web Mining, SIAM Conference on Data Mining*, 33-40.

Bar-Yossef, Z., & Rajagopalan, S. (2002). Template detection via data mining and its applications. *Proceedings of the 11th International World Wide Web Conference (WWW2002)*, 580-591.

Cadez, I. V., Heckerman, D., Meek, C., Smyth, P., & White, S. (2003). Model-based clustering and visualization of navigation patterns on a Web site. *Journal of Data Mining and Knowledge Discovery, 7*(4), 399-424.

Catledge, L., & Pitkow, J. (1995). Characterizing browsing behaviors on the World Wide Web. *Computer Networks and ISDN Systems, 6*(27), 1065-1073.

Chakrabarti, S. (2003). *Mining the Web.* San Francisco: Morgan Kaufmann.

Chen, K., & Liu, L. (2003). Validating and refining clusters via visual rendering. *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003)*, 501-504.

Chen, M. S., Park, J. S., & Yu, P. S. (1998). Efficient data mining for path traversal patterns. *IEEE Transactions on Knowledge and Data Engineering, 10*(2), 209-221.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Cooley, R., Mobasher, B., & Srivastava, J. (1999). Data preparation for mining World Wide Web browsing patterns. *Knowledge Information Systems, 1*(1), 5-32.

Dempster, A. P., Laird, N. P., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B, 39*, 1-22.

Eiron, N., & McCurley, K. S. (2003). Untangling compound documents on the Web. *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, 85-94.

Flake, G. W., Tarjan, R. E., & Tsioutsiouliklis, K. (2004). Graph clustering and minimum cut trees. *Internet Mathematics, 1*(4), 385-408.

Fraley, C., & Raftery, A. (1998). How many clusters? Which clustering method? Answers via model-based cluster analysis. *Computer Journal, 41*, 578-588.

Fu, Y., Sandhu, K., & Shih, M. Y. (1999). A generalization-based approach to clustering of Web usage sessions. In *Proceedings of the International Workshop on Web Usage Analysis and User Profiling (WEBKDD1999)* (LNCS 1836, pp. 21-38). San Diego: Springer Verlag.

Goker, A., & He, D. (2000). Analysing Web search logs to determine session boundaries for user-oriented learning. In *Proceedings of the International Conference of Adaptive Hypermedia and Adaptive Web-Based Systems (AH2000)* (LNCS 1892, pp. 319-322). Trento, Italy: Springer Verlag.

Gomory, S., Hoch, R., Lee, J., Podlaseck, M., & Schonberg, E. (1999). Analysis and visualization of metrics for online merchandising. In *Proceedings of the International Workshop on Web Usage Analysis and User Profiling (WEBKDD1999)* (LNCS 1836, pp. 126-141). San Diego, CA: Springer Verlag.

Greco, G., Greco, S., & Zumpano, E. (2004). Web communities: Models and algorithms. *World Wide Web Journal, 7*(1), 58-82.

Günter, S., & Bunke, H. (2003). Validation indices for graph clustering. *Pattern Recognition Letters, 24*(8), 1107-1113.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002a). Cluster validity methods: Part I. *SIGMOD Record, 31*(2), 40-45.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002b). Cluster validity methods: Part II. *SIGMOD Record, 31*(3), 19-27.

Hammouda, K. M., & Kamel, M. S. (2004). Efficient phrase-based document indexing for Web document clustering. *IEEE Transactions on Knowledge Data Engineering, 16*(10), 1279-1296.

Huang, X., Peng, F., An, A., & Schuurmans, D. (2004). Dynamic Web log session identification with statistical language models. *Journal of the American Society for Information Science and Technology (JASIST), 55*(14), 1290-1303.

Huang, Z., Ng, J., Cheung, D. W., Ng, M. K., & Ching, W. (2001). A cube model for Web access sessions and cluster analysis. In *Proceedings of the International Workshop on Web Usage Analysis and User Profiling (WEBKDD2001)* (LNCS 2356, pp.48-67). Hong Kong, China: Springer Verlag.

Huang, Z., Ng, M. K., & Cheung, D. (2001). An empirical study on the visual cluster validation method with fastmap. *Proceedings of the 7th International Conference on Database Systems for Advanced Applications (DASFAA 2001)*, 84-91.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Ino, H., Kudo, M., & Nakamura, A. (2005). Partitioning of Web graphs by community topology. *Proceedings of the 14th International Conference on World Wide Web (WWW 2005)*, 661–669.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys, 31*(3), 264-323.

Jushmerick, N. (1999). Learning to remove Internet advertisements. *Proceedings of the 3rd Annual Conference on Autonomous Agents*, 175-181.

Lian, W., Cheung, D. W., Mamoulis, N., & Yiu, S. (2004). An efficient and scalable algorithm for clustering XML documents by structure. *IEEE Transactions on Knowledge Data Engineering, 16*(1), 82-96.

Lou, W., Liu, G., Lu, H., & Yang, Q. (2002). Cut-and-pick transactions for proxy log mining. *Proceedings of the 8th International Conference on Extending Database Technology (EDBT 2002)*, 88-105.

Masada, T., Takasu, A., & Adachi, J. (2004). Web page grouping based on parameterized connectivity. *Proceedings of the 9th International Conference on Database Systems for Advanced Applications (DASFAA 2004)*, 374-380.

Mobasher, B., Cooley, R., & Srivastava, J. (2000). Automatic personalization based on Web usage mining. *Communications of the ACM, 43*(8), 142-151.

Modha, D., & Spangler, W. (2003). Feature weighting in *k*-means clustering. *Machine Learning, 52*(3), 217-237.

Moore, J., Han, E., Boley, D., Gini, M., Gross, R., Hastings, K., et al. (1997). Web page categorization and feature selection using association rule and principal component clustering. Proceedings of the *7th Workshop on Information Technologies and Systems*, Atlanta, GA.

Nanopoulos, A., Katsaros, D., & Manolopoulos, Y. (2003). A data mining algorithm for generalized Web prefetching. *IEEE Transactions on Knowledge Data Engineering, 15*(5), 1155-1169.

Nierman, A., & Jagadish, H. V. (2002). Evaluating structural similarity in XML documents. *Proceedings of the 5th International Workshop on the Web and Databases (WebDB 2002)*, 61-66.

Pallis, G., Angelis, L., & Vakali, A. (2005). Model-based cluster analysis for Web users sessions. In *Proceedings of the 15th International Symposium on Methodologies for Intelligent Systems (ISMIS 2005)* (LNCS 3488, pp. 219-227). Saratoga Springs, NY: Springer Verlag.

Pallis, G., Angelis, L., Vakali, A., & Pokorny, J. (2004). A probabilistic validation algorithm for Web users' clusters. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC 2004)*, 4129-4134.

Pallis, G., & Vakali, A. (2006). Insight and perspectives for content delivery networks. *Communications of the ACM, 49*(1), 101-106.

Pallis, G., Vakali, A., Angelis, L., & Hacid, M. S. (2003). A study on workload characterization for a Web proxy server. *Proceedings of the 21st IASTED International Multi-Conference on Applied Informatics (AI 2003)*, 779-784.

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

Reddy, P. K., & Kitsuregawa, M. (2001). An approach to relate the Web communities through bipartite graphs. *WISE, 1*, 301-310.

Salton, G., Wong, A., & Yang, C. (1975). A vector space model for automatic indexing. *Communications of the ACM, 18*(11), 613-620.

Scherbina, A., & Kuznetsov, S. (2004). Clustering of Web sessions using Levenshtein metric. In *Proceedings of the 4th Industrial Conference on Data Mining (ICDM 2004)* (LNCS 3275, pp. 127-133). San Jose, CA: Springer Verlag.

Shahabi, C., Faisal, A., Kashani, F. B., & Faruque, J. (2000). INSITE: A tool for interpreting users' interaction with a Web space. *Proceedings of the 26th International Conference on Very Large Data Bases (VLDB 2000)*, 635-638.

Shahabi, C., Zarkesh, A. M., Adibi, J., & Shah, V. (1997). *Knowledge discovery from users Web page navigation.* Proceedings of the 7th International Workshop on Research Issues in Data Engineering (IEEE RIDE), Birmingham, United Kingdom.

Smith, K., & Ng, A. (2003). Web page clustering using a self-organizing map of user navigation patterns. *Decision Support Systems, 35*(2), 245-256.

Spiliopoulou, M., & Faulstich, L. (1998). WUM: A tool for WWW utilization analysis. In *Proceedings of the International Workshop on World Wide Web and Databases (WebDB 1998)* (LNCS 1590, pp. 184-203). Valencia, Spain: Springer Verlag.

Stein, B., Eissen, S. M., & Wibrock, F. (2003). *On cluster validity and the information need of users.* Proceedings of the 3rd IASTED International Conference on Artificial Intelligence and Applications (AIA 2003), Benalmadena, Spain.

Vel, O. D., Anderson, A., Corney, M., & Mohay, G. (2001). Mining e-mail content for author identification forensics. *Special Interest Group on Management of Data Record (SIGMOD Rec.), 30*(4), 55-64.

Vesanto, J., & Alhoniemi, E. (2000). Clustering of self-organizing map. *IEEE Transactions on Neural Networks, 11*(3), 586-600.

Wang, W., & Zaïane, O. R. (2002). Clustering Web sessions by sequence alignment. In *Proceedings of the 13th International Workshop on Database and Expert Systems Applications (DEXA 2002)* (LNCS 2453, pp. 394-398). Aix-en-Provence, France: Springer Verlag.

Wen, J. R., Nie, J. Y., & Zhang, H. (2001). Clustering user queries of a search engine. *Proceedings of the 10th International World Wide Web Conference (WWW2001)*, 162-168.

Wong, W., & Fu, A. (2000). *Incremental document clustering for Web page classification.* Proceedings of the IEEE International Conference on Information Society in the 21st Century: Emerging Technologies and New Challenges (IS2000), Fukushima, Japan.

Wu, K., Yu, P. S., & Ballman, A. (1998). Speedtracer: A Web usage mining and analysis tool. *IBM Systems Journal, 37*(1), 89-105.

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. *Proceedings of the 14th International Conference on Machine Learning (ICML 1997)*, 412-420.

Zaïane, O. R., Foss, A., Lee, C.-H., & Wang, W. (2002). On data clustering analysis: Scalability, constraints, and validation. In *Proceedings of the 6th Pacific-Asia Conference*

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

of Advances in Knowledge Discovery and Data Mining (PAKDD 2002) (LNCS 2336, pp. 28-39). Taipei, Taiwan: Springer Verlag.

Zamir, O., Etzioni, O., Madanim, O., & Karp, R. M. (1997). Fast and intuitive clustering of Web documents. *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining* (*KDD 1997)*, 287-290.

Zeng, H. J., He, Q. C., Chen, Z., Ma, W. Y., & Ma, J. (2004). Learning to cluster Web search results. *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 210-217.

Zhu, J., Hong, J., & Hughes, J. (2004). PageCluster: Mining conceptual link hierarchies from Web log files for adaptive Web site navigation. *ACM Transactions on Internet Technologies, 4*(2), 185-208.

Endnotes

1 http://www.squid-cache.org/Scripts

2 http://www.w3.org/

3 The term proximity is often used as a general term to denote either a measure of similarity or dissimilarity.

4 In a hard assignment policy, each object is assigned to only one cluster. On the other hand, a soft assignment policy allows degrees of membership in multiple clusters, which means that one object can be assigned to multiple clusters with certain membership values.

5 http://www-users.cs.umn.edu/~karypis/cluto/index.html

6 http://www.mindlab.de

7 http://www.cs.wisc.edu/niagara/Introduction.html

8 http://popfile.sourceforge.net

9 http://www.akamai.org

10 http://www.limelightnetworks.com/

11 http://www.mirror-image.com

Copyright © 2007, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.