The Scent of a Newsgroup: Providing Personalized Access to Usenet Sites through Web Mining*

Giuseppe Manco[1], Riccardo Ortale[2], and Andrea Tagarelli[2]

[1] ICAR-CNR – Institute of Italian National Research Council
Via Bucci 41c, 87036 Rende (CS), Italy
e-mail: manco@icar.cnr.it

[2] DEIS, University of Calabria
Via Bucci 41c, 87036 Rende (CS), Italy
e-mail: {ortale, tagarelli}@si.deis.unical.it

Abstract. We investigate Web mining techniques focusing on a specific application scenario: the problem of providing personalized access to Usenet sites accessible through a Web server. We analyze the data available from a Web-based Usenet server, and describe how traditional techniques for pattern discovery on the Web can be adapted to solve the problem of restructuring the access to news articles. In our framework, a personalized access tailored to the needs of each single user, can be devised according to both the content and the structure of the available data, and the past usage experience over such data.

1 Introduction

The wide exploitation of new techniques and systems for generating and storing data has made available a huge amount of information, which can be profitably used in the decision processes. Such volumes of data highlight the need for analysis methodologies and tools. The term "Knowledge Discovery in Databases" is usually devoted to the (iterative and interactive) process of extracting valuable patterns from the data by exploiting *Data Mining* algorithms. In general, data mining algorithms find hidden structures, tendencies, associations and correlations among data, and mark significant information.

An example of data mining application involving huge volumes of data is the detection of behavioral models on the Web. Typically, when users interact with a Web service (available from a Web server), they provide enough information on their requirements: what they ask for, what they do not ask for, which experience they gain in using the service, how they interact with the service itself. The possibility of tracking users' browsing behavior, in terms of both the individual Web pages visited and her/his on-line transactions,

* This work was partially supported by the National Research Council project SP2: "Strumenti, ambienti e applicazioni innovative per la società dell'informazione - Legge 449/97-99".

offers a new perspective of interaction between service providers and end users. Indeed, the above scenario is only one possible application of Web mining techniques, which consists in applying data mining algorithms to discovery patterns from Web data. A classification of Web mining techniques can be devised in three main categories:

- *Structure mining*. It is intended here to infer information from the topology of the link structure among Web pages (Dhyani et al., 2002). This kind of information is useful for a number of purposes: categorization of Web sites, gaining an insight in the similarity relations among Web sites, developing suitable metrics for the evaluation of the relevance of Web pages.
- *Content mining*. The main aim is to extract useful information from the content of Web resources (Kosala and Blockeel, 2000). Content mining techniques can be applied to data of different kinds: unstructured, semistructured (such as HTML and XML documents), structured (such as relational tables and digital libraries), dynamic (such as responses to database queries). Content mining methodologies are also related to traditional Information Retrieval techniques (Baeza-Yates and Ribeiro-Neto, 1999). However, the application and extension of such techniques to Web resources allows the definition of new challenging application domains (Chakrabarti, 2002): Web query systems, which exploit information about the structure of Web documents to handle with complex search queries; intelligent search agents, which work on behalf of users based both on a description of their profile and a specific domain knowledge for suitably mining the results that search engines provide in response to user queries.
- *Usage mining*. The focus here is the application of data mining techniques to discover usage patterns from Web data (Srivastava et al., 2000), in order to understand and better serve the needs of Web-based applications and end users. Web access logs are the main data source for any Web usage mining activity: data mining algorithms can be applied to such logs in order to infer information describing the usage of Web resources. The analysis of Web usage can be profitably applied to devise intelligent strategies for caching and prefetching Web resources within either Web or proxy servers, automatically improve the link structure of Web sites (adaptive Web sites), recommendation systems and customer management in an e-business context (Perkowitz and Etzioni, 2000; Mobasher et al., 2000).

Web-based information systems depict a typical application domain for the above Web mining techniques, since they allow the user to choose contents of interest and browse through such contents. As the number of potential users progressively increases, a large heterogeneity in interests and in the knowledge of the domain under investigation is exhibited. Therefore, a Web-based information system must tailor itself to different user requirements, as well as to different technological constraints, with the ultimate aim of personalizing and improving user's experience in accessing the system. In general, an adaptive Web-based information system is based on a suitable model of *user behavior*, a model of the *application domain* (i.e., the identification of the components of a Web system which can be adaptively deployed), and a model of the *adaptation process* (i.e., the identification of suitable personalization rules which enable adapting both content and usage). It is

clear from this context how Web mining techniques can be leveraged to deploy such models and contribute in the definition of a highly effective adaptive system.

The main objective of this chapter is to analyze the depicted Web mining techniques according to a specific application scenario, which suitably combines all the above issues in an overall framework: the problem of providing personalized access to the contents of Usenet communities available from the Web. We refer mainly to the possibility of accessing newsgroups from a Web interface (such as, e.g., in `groups.google.com`), and envisage a Web-based service capable of providing personalized access by exploiting Web mining techniques.

A Web-enabled Usenet access through a Web server has some major advantages w.r.t. the traditional access provided by NNTP (Net News Transfer Protocol) servers. In particular, from a user point of view, a Web-based service has the advantage of offering ubiquitous and anytime access through the World-Wide Web, without the need of having a predefined client other than a Web browser. In addition, a user can fruitfully exploit further consolidated Web-based services (such as, e.g., search engine capabilities). From a service provider point of view, it offers the possibility of tracking a given user in her/his interaction with the service for free, by exploiting traditional Web-based techniques: for example, by analyzing server logs, one can understand when users access the service, which newsgroups they are interested in, which topics are more appealing in the various communities. It is clear that such an infrastructure can substantially benefit from the analysis f both the content, the structure and the usage of the available data.

The chapter is organized as follows. In Section 2 a detailed description of the data sources available from a Web-based Usenet service is given. Section 3 provides a rationale for the inadequacy of current Usenet services, and introduces the main lines of investigation which are analyzed in the subsequent sections. These are: content mining (Section 4), in which particular emphasis is posed to topic discovery and maintenance; usage mining (Section 5), where data preprocessing and cleaning plays a crucial role; and structure mining (Section 6), where the structure of usenet news is investigated and interesting applications are devised on the basis of such a structure. Finally, section 7 provides an overview of personalization techniques and describes a specific application of a well-known algorithm to the case of Usenet access.

2 The Information Content of a Usenet Site

Usenet operates in a peer-to-peer framework which allows the users to exchange public messages on a wide variety of topics including computers, scientific fields, politics, national cultures, and hobbies. Differently from email messages, Usenet articles are concerned with public discussions rather than personal communications and are grouped, according to their main subject, into newsgroups. Practically speaking, newsgroups are collections of articles sharing the same topics.

Newsgroups can be organized into hierarchies of topics. Information about the hierarchy can be found in the newsgroup names themselves. Indeed, newsgroup names generally contain two or more parts, separated by periods. The first part of the name indicates the top-level hierarchy to which the newsgroup belongs (the

standard "Big Seven" top-level hierarchies are: `comp`, `misc`, `news`, `rec`, `sci`, `soc`, `talk`). Reading from left to right, the various parts of the name progressively narrow the topic of discussion. For example, the newsgroup `comp.lang.java.programmer` contains articles discussing programming issues concerning the Java programming language in Computer Science. Notice that, while newsgroup hierarchy is highly structured, articles can be posted to multiple newsgroups. This usually happens when articles contain more than one topic of discussion: for example, an article concerning the use of an *Oracle JDBC Driver* can be posted to both `comp.lang.java.programmer` and `comp.databases.oracle`. Figure 1 exemplifies the hierarchy of the newsgroups, and the membership links between articles and newsgroups.

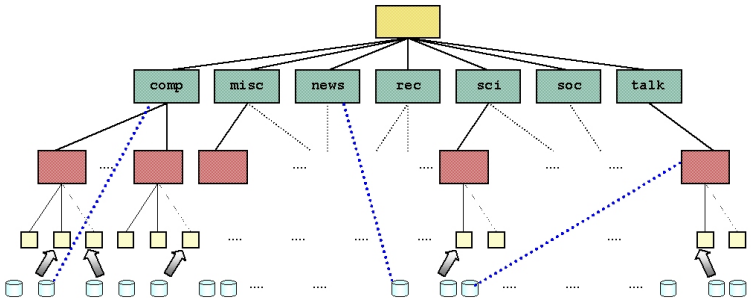

Fig. 1. Hierarchy of newsgroups

The format for Usenet articles was conceived to fit in with existing tools for managing messages in Internet. The Internet standard format RFC-822 for mail messages meets most of the needs of Usenet, therefore Usenet articles must be formatted as valid Internet mail messages. On the other hand, additional requirements on each message have to be placed while some Internet features have to be forbidden. The result is that the Usenet news standard (RFC-1036) is more restrictive than the Internet standard.

In our scenario, Usenet articles can be accessed by means of a Web server, which allows both the visualization of articles, and the navigation of newsgroup hierarchies. An example such interface is provided in Figure 2. Users can navigate the hierarchy of newsgroups by accessing highly structured pages, in which hyperlinks lead to specific portions of the hierarchy. For each newsgroup, both the sub-hierarchy and the messages available can be displayed. Articles can be further grouped in threads, where a thread contains an initial article, representing an argument of discussion raised by some user, and a chain of answers to the initial article.

By a closer look at the overall architecture of the system (depicted in Figure 3), two main information sources can be detected: *Access logs*, describing users' browsing behavior, and *Usenet repositories*, representing the articles a user may access. We now analyze such information sources in deeper details.

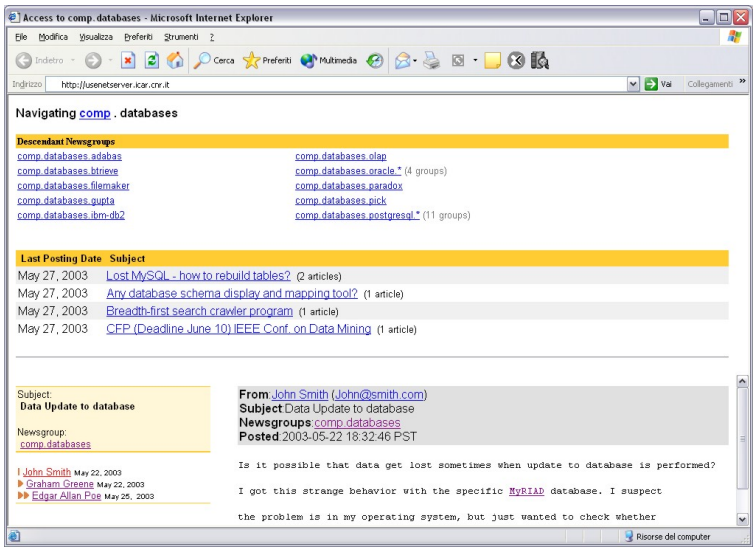

Fig. 2. An example Web interface to Usenet news

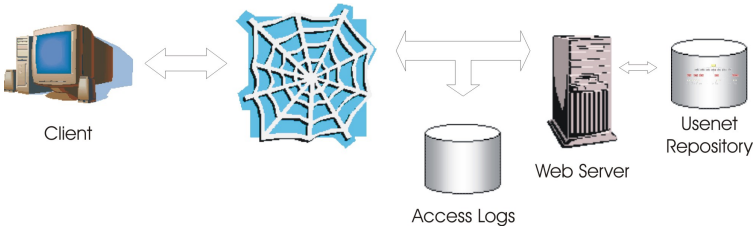

Fig. 3. Architecture of a Web-based Usenet server

2.1 Web logs: a mechanism for collecting user clickstream

Usage data are at the basis of any Web usage mining process. These data can be collected at different levels on the ideal communication channel between a generic user and the Web site currently accessed: at the client level, proxy-server level, Web site level. Sources at different levels take into account different segments of Web users and, as a consequence, highlight distinct browsing patterns (Cooley, 2000). Precisely, sources at the client level focus on *single user/single site* (or even *single user/multi site*) browsing behavior. Information related to *multi user/multi site* navigational behavior are collected at the proxy-server level. Finally, sources at the Web site level reveal usage patterns of type *multi user/single site*. Since the user profiling phase aims at gaining an insight into the browsing strategies of visitors, we focus on the sources of usage data at Web site level.

Web logs represent a major approach to tracking user behavior. By recording any incoming request to a Web server, Web logs explicitly capture visitors' browsing behavior in a concurrent and interleaved manner. A variety of ad-hoc formats have been devised for organizing data within Web logs: two popular formats are CLF (Common Log Format) and ECLF (Extended Common Log Format). Recently introduced by the

W3C, ECLF improves CLF by adding a number of new fields to each entry, which reveal particularly useful for demographic analysis and log summaries (Eirinaki and Vazirgiannis, 2003).

A concise description of the fields in an ECLF format follows. *IP Address* is the Internet address of the remote host from which a request originated: it may be a proxy-server address. *User ID* is filled in only in those cases in which users are requested to provide their authentication to access secure data on the Web server. *Time* is the time stamp which indicates when an incoming request was received by the Web server. The field *Request* indicates the main components of a request, namely its method (typically GET, POST or HEAD), the URI (Uniform Resource Identifier) of the required resource and the protocol of the request itself (typically, HTTP). *Status* is exploited to record the kind of response to a particular request: it may be a redirection, a successful delivery of the required resource, an internal Web server error, or an error during the delivery process. *Size* reflects the amount of bytes of a response to a user request. *Referrer* is the URI of the Web resource from which the incoming request originated. Finally, *Agent* is a field which gives details about the nature of the operating system and the browser exploited at the client level. ECLF logs can also include cookies, i.e. pieces of information uniquely generated by Web servers to address the challenging task of identifying and tracking users during their browsing activities.

The main limitation which affects Web logs is that they do not allow to reliably capture user behavior. For instance, the viewing time perceived at the Web site level may be much longer than it is actually at the client level. This is typically due to a number of unavoidable reasons, such as the client bandwidth, the transmission time necessary for the Web server to deliver the required resource and the congestion status of the network. Also, Web logs may not record some user requests. Such a loss of information usually happens when a user repeatedly accesses a same page and caching is present. Typically, only the first request is captured: subsequent requests may be served by a cache, which can be either local to the client or part of an intermediate proxy-server. These drawbacks must be taken into account while extracting usage pattern from Web logs: a variety of heuristics have been devised in order to preprocess Web logs in a such a way to reduce the side effects of the above issues.

Data about user behavior can also be collected through alternative approaches (Cooley, 2000), which allow to overcome a traditional limitation which affects the exploitation of Web logs: the impossibility of capturing information other than that in the HTTP header of a Web request. Among them, we mention ad-hoc tracking mechanisms, which allow to define meaningful application-dependent logs describing user browsing activities at the required degree of detail. Indeed, the exploitation of new technologies for developing Web (application) servers (such as, e.g., JSP/PHP/.NET frameworks) offers great opportunities to control both the delivery and tracking of the information requested by a given user. Two essential features characterize the process of usage data collection within an application server.

First, it can capture those information which are not typically addressed by Web logs such as the request parameters sent to the Web server through the hidden POST method and the state variables of an application (such as the values stored in a user session). Also, it can take into account a number of meta information

concerning the contents and the structure of a given Web site, the specific parameters behind general queries to content databases exploited to dynamically generate Web pages, and, for each Web page, its actual size and last modified time.

Second, it guarantees the reliability of the information recorded. When application servers are part of the infrastructure of a Web site, it may happen that the *Size* field in a Web log is unreliable. For instance, a dynamic Web page may be served by an application server to a given user. In such cases, the Web server considers logs a success in the entry associated to the user request. However, such a value can be unreliable: this typically happens when the responses correspond to pages with messages detailing some internal error, which makes the delivery of the original request unfeasible. Moreover, the possibility of logging specific parameters of user requests allows to keep trace of the identity of each user. Registration typically requires any user to provide her/his own uniquely-assigned password before accessing a Web site: henceforth, all the successive requests, originated by a same password, will be reliably associated to that user.

A format for the generic entry of an application log, suitably thought for the Usenet environment, can be devised. It may consists of (at least) three fields. *User* is a fields which refers to the identity of the user who made the request: it can be either an IP address or a unique user-id. *Time* is a time stamp for the request. *Request* logs the details (such as the level in the newsgroup hierarchy, or the topics) of the request.

2.2 Content data: finding structure within news articles

News articles are the primary information source of Usenet. Differently from generic text documents (to which at a first sight they could be assimilated), some relevant features properly characterize them:

- Articles usually may have a rather short size. This may look as a sociological observation, since the size of an article depends on a set of factors such as topic involved, user preferred behavior, degree of interaction required. However, newsgroups are conceived as a means for generating discussions, which typically consist in focused questions and (rather) short replies. Some statistics computed over the articles available in some newsgroups, revealed a typical average article size of 150 words. By contrast, the TREC document collection (generally used as a standard text mining benchmark (Baeza-Yates and Ribeiro-Neto, 1999)) provides documents with an average size of 860 words.
- Text documents usually exhibit only unstructured features (words occurring in the document). By contrast, news articles result in a combination of important structural properties (such as sender, recipients, date and time, and whether the article represents a new thread or a reaction in a given thread), and unstructured components (*Subject*, *Content*, *Keywords* headers).

In the following, we briefly review how to extract relevant information from a collection of news articles. This requires a study on the representation of both structured and unstructured features of a news article.

Concerning text contents, the extraction of relevant features is usually performed by a sequence of well-known text operations (Moens, 2000; Baeza-Yates and Ribeiro-Neto, 1999), such as lexical analysis,

removal of stopwords, lemmatization and stemming. The above text operations associate each article with a set of terms that are assumed to reflect at best the textual content of the article. However, such terms have different discriminating power, i.e., its relevance in the context where it is used. Many factors may contribute in index term weighting: statistics on the text (e.g., size, number of different terms appearing in), relationships between an index term and the document containing it (e.g., location, number of occurrences), and relationships between an index term and the overall document collection (e.g., number of occurrences).

A commonly used weighting function is based on the finding that the most significant terms are those occurring frequently within a document, but rarely within the remaining documents of the collection. To this purpose, the weight of an index term can be described as a combination of its frequency of occurrence within a document (*Term Frequency - TF*) and its rarity across the whole collection (*Inverse Document Frequency - IDF*). A widely used model complying with the above notions is the *vector-space model* (Baeza-Yates and Ribeiro-Neto, 1999), in which each article is represented as a n-dimensional vector $\mathbf{w}$, where n is the number of available index terms and each component w_j is the (normalized) *TF.IDF* weight associated with index term j:

$$w_{ji} = \frac{tf_{ji} \cdot idf_j}{\sqrt{\sum_p (tf_{pi} \cdot idf_p)^2}}$$

It is well-known from the literature that this model tends to work quite well in practice despite a number of simplifying assumptions (e.g., assumption of term independence, absence of word-sense information, as well as of phrase-structure and word-order).

feature	**type**	**source header**
Newsgroup hierarchy (e.g., `comp.lang.c`)	categorical	*Newsgroups:*
Followup newsgroup hierarchy	categorical	*Followup-To:*
Sender domain (e.g., `yahoo.com`)	categorical	*From:*
Weekday	categorical	*Date:*
Time period (e.g., early morning, afternoon, evening)	categorical	*Date:*
Expiration date	categorical	*Expires:*
Geographic distribution (e.g., `world`)	categorical	*Distribution:*
Article length	numeric	*Lines:*
Nr. of levels in newsgroup hierarchy	numeric	*Newsgroups:*

Table 1. Structured features of news articles

From each article, further features can be extracted from both the required headers (such as *From*, *Date*, *Newsgroups* and *Path*) and the optional headers (such as *Followup-To*, *Summary*, *References*). Indeed, discriminant information can be obtained by exploiting the hierarchy of topics inferred by newsgroups, or by analyzing the temporal shift of articles for a given topic. Notice also that, in principle, textual contents may

contain further information sources, such as, e.g., hyperlinks. To this purpose, it is particularly interesting to notice that, a Web interface to a usenet service assigns each article a unique URL address, thus allowing articles to link each other directly by means of such URLs.

Summarizing, an article can be represented by feature vector $\mathbf{x} = (\mathbf{y}\ \mathbf{w})$ in which the structured component (denoted by $\mathbf{y}$) may comprise, e.g., the features reported in Table 1, while unstructured information (denoted by $\mathbf{w}$) is mainly obtained from the content and from the *Subject* header (or from the *Summary* and *Keywords* headers as well).

3 Providing Personalized Access to News Articles

Usenet encompasses a very large community including government agencies, large universities, high schools, businesses of all sizes. Here, newsgroups on new topics are continuously generated, new articles are continuously posted, and (new) users continuously access the newsgroups looking for articles of interest. According to this point of view, Usenet can be thought as a medium generating (at least) two main continuous streams, which any management system should account for: one stream for article contents, and another stream for article accesses.

In such a context, the idea of providing personalized access to the content of Usenet articles is quite attractive, for a number of reasons.

1. First of all, the hierarchy provided by the newsgroups is often inadequate, since too many newsgroups deal with overlapping subjects, and even a single newsgroup dealing with a specific topic may contain rather heterogenous threads of discussion. As an example, the newsgroup `comp.lang.java.programmer` (which should deal with programming issues in the Java programming language) contains threads which can be grouped in different subtopics, dealing respectively with "typing", "networking", "debugging", etc. As a consequence, since the size of the newsgroups dynamically grows, when looking for answers to a specific query one can frequently incur in the *abundance problem*, which happens when too many answers are available and the degree of relevance of each answer has to be suitably weighted. By contrast, one can even incur in the *scarcity problem*, which usually happens when the query is too specific and as a consequence many viable answers are missed. Hence, articles in the hierarchy provided by the newsgroups needs better (automatic) organization according to contents of articles, in order to facilitate search and detection of relevant information.
2. Answers to specific threads, as well as references to specific articles, can be analyzed from a *structural* point of view. For example, the graph structure of the accesses to articles (available from users' access logs) can be investigated, thus allowing the identification of *hub* and *authoritative* articles, as well as users with specific areas of expertise. Thus, the analysis of the graphs devised from both accesses and reactions to specific articles is clearly of greater impact to the purpose of providing personalized access the Usenet service.

3. Since users can be tracked, their preferences, requirements and experiences in accessing newsgroups can be evaluated directly from the access logs. As a consequence, both available contents and their presentation can be adapted according to user's profile, which can be incrementally built as soon as the user provides sufficient information about her/his interaction with the service. Moreover, the experience provided by the interaction of a given user can be adapted to users exhibiting a similar profile, thus enabling a *collaborative system* in which experiences are shared among users.

The problem of providing personalized access to the Usenet communities available from the Web can be ideally divided into three main phases: *user profiling*, the process of gaining an insight about the preferences and tastes of Usenet visitors through an in-depth analysis of their browsing behavior; *content profiling*, the process of gaining an insight about the main issues and topics appearing in the content and in the structure of the articles; and *personalization*, the adoption of ad-hoc strategies to tailor the delivery of Usenet contents to a specific profile.

4 Mining the Content of News Articles

A simple and straightforward personalization strategy consists in providing searching capabilities within the Usenet news. Indeed, many Web-based Usenet servers (such as Google) provide keyword-based querying capabilities, and ranking mechanisms for the articles in the Usenet (or in a specific newsgroup) w.r.t. the query of interest. In general, queries involve relationships between terms and documents, such as "find the articles dealing with Java Native Interface", and hence can be modeled as vectors in the vector-space model. A typical ranking mechanism hence may consist in computing the score of each article in a collection as its *cosine similarity* of its representative vector $\mathbf{w}$ with the vector q representing the query: $r_{q,\mathbf{w}} = \mathbf{w} \cdot q$. Articles with highest rank constitute the answer of the query.

A further interesting direction is the identification of the topics emerging from the articles. This can be accomplished mainly by means of clustering of articles on the basis of their contents. Clustering aims at identifying homogeneous groups that shall represent sub-newsgroups in the re-organized news collection. Formally, the problem can be stated as follows: given a set $\mathcal{M} = \{m_1, \ldots, m_N\}$ of news articles, we aim at finding a suitable partition $\mathcal{P} = \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ of $\mathcal{M}$ in k groups, such that each group contains an homogeneous subset of articles (or threads), with an associated label describing the main topics the group deals with. The identification of homogeneous groups relies on the capability of: *(i)* defining matching criteria for articles according to their contents; *(ii)* detecting representative descriptions for each cluster; and *(iii)* exploiting suitable clustering schemes.

Homogeneity can be measured by exploiting the feature vectors defined in Section 2.2. A similarity measure $s(\mathbf{x}_i, \mathbf{x}_j)$ as a real number, is defined as $s(\mathbf{x}_i, \mathbf{x}_j) = \alpha s_1(\mathbf{y}_i, \mathbf{y}_j) + (1-\alpha) s_2(\mathbf{w}_i, \mathbf{w}_j)$, where s_1 defines the similarity of the structured component of the articles, and s_2 takes into account the unstructured part of the articles. In particular, s_1 can be defined by resorting to traditional similarity measures, such as *dice*,

euclidean or *mismatch-count* distance (Huang, 1998). Mismatch-count distance could be exploited, e.g., by taking into account the hierarchy of subgroups, as in principle articles posted in newsgroups sharing many levels in the newsgroup hierarchy are more likely to be similar than articles posted in newsgroups sharing few levels. On the other way, s_2 can be chosen among the similarity measures particularly suitable for documents (Strehl et al., 2000; Baeza-Yates and Ribeiro-Neto, 1999), such as the *cosine coefficient*. Finally, $\alpha \leq 1$ represents the weight to be associated with the structured component. As shown in (Manco et al., 2002), suitable values can be obtained by a detailed analysis of the documents in the collection.

Concerning labels, we have to find a suitable way of associating to a given group a significant label. The structural part is simple to deal with, since we can associate the *mode* vector (Huang, 1998) with structured components $\mathbf{y}$. The unstructured part requires some attention. In a sense, we are looking for a label reflecting the content of the articles within the cluster, and contemporarily capable of distinguishing two different clusters. To this purpose, a simple strategy could be to resort to *frequent itemsets discovery* techniques (Agrawal and Srikant, 1994; Beil et al., 2002), and associate to each cluster the sets of terms (or concepts) which more frequently appear together within the clusters.

Many different clustering algorithms can be exploited (Jain et al., 1999) to cluster articles according to the above mentioned matching criteria. Hierarchical methods are widely known as providing clusters with a better quality (Baeza-Yates and Ribeiro-Neto, 1999; Steinbach et al., 2000). In the context of newsgroup mining, the adoption of hierarchical approaches is particularly attractive, since it should allow the generation of a hierarchy of newsgroups. Hierarchical approaches suffer of serious efficiency drawbacks, since they require quadratic time complexity in the number of articles. By contrast, efficient centroid-based methods have been proposed in the literature. In particular, (Dhillon and Modha, 2001) define a suitable partitional technique, namely *spherical k-Means*, that has the main advantage of requiring $O(N)$ comparisons and guaranteeing a good quality of clusters. It follows from Section 2.2 that feature vectors $\mathbf{w}_i$ are points within the unit sphere. In such cases, the computation of the cosine similarity is reduced to the computation of the scalar product among two vectors. The spherical k-Means algorithm aims at maintaining such a property during the whole clustering phase. For each cluster C_j containing n_j documents, the algorithm computes the cluster center $\mu_j = 1/n_j \sum_{\mathbf{w} \in C_j} \mathbf{w}$. By normalizing μ_j, we obtain the *concept vector* $\mathbf{c}_j$ of C_j, which is the feature vector that is closest in cosine similarity to all the document vectors in the cluster C_j.

The main drawback of centroid-based techniques is that the quality of their result is strictly related to two main issues: the number of desired clusters, which has to be known a priori, and the choice of a set of suitable initial points. Combinations of hierarchical agglomeration with iterative relocations can be devised here, by first using the hierarchical agglomerative algorithm over a small arbitrary subset $\mathcal{S}$ of articles to seed the initial clusters for k-Means methods. Figure 4 shows an example such integration. Practically, by choosing a subset $\mathcal{S}$ of $\mathcal{M}$, with a size $h \ll N$ (e.g., $h = \sqrt{kN}$), as an input for a hierarchical clustering scheme, we avoid the related efficiency issues. The resulting partition provided by such an algorithm can be exploited within the k-Means algorithm, as it provides an optimal choice for both the desired number of

Input : A set $\mathcal{M} = \{m_1, \ldots, m_N\}$ of news articles.
Output : A partition $\mathcal{P} = \{C_1, \ldots, C_k\}$ of $\mathcal{M}$.
Method :

- Sample a small subset $\mathcal{S} = \{m_1, \ldots, m_h\}$ of news articles randomly chosen from $\mathcal{M}$.
- Obtain the feature vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_h\}$ from $\mathcal{S}$.
- Apply hierarchical agglomerative algorithm on $\{\mathbf{x}_1, \ldots, \mathbf{x}_h\}$.
- Let $\mathbf{c}_1, \ldots, \mathbf{c}_k$ be the concept vectors representing the clusters of the partition supplied from the hierarchical algorithm:
  - Apply the k-Means algorithm on $\mathcal{M}$ exploiting $\mathbf{c}_1, \ldots, \mathbf{c}_k$ as initial centers.
  - Answer $\mathcal{P} = \{C_1, \ldots, C_k\}$ as the result of the k-Means algorithm.

Fig. 4. An example integration of hierarchical and partitional clustering

clusters and the initial cluster centers (computed starting from the partition provided by the hierarchical approach).

4.1 The problem of posting incoming news articles

Clearly, a personalized Usenet server should take into account the dynamic nature of the service: continuously, new newsgroups are generated, and new articles are posted to newsgroups. In the context of topic discovery, an interesting exploitation of data mining techniques can be the classification of incoming articles according to their contents. The problem of classifying incoming articles can to be tackled according to two main perspectives:

- Articles posted to a newsgroup could be of interest to other newsgroups, too. Here, routing techniques should be devised, mainly based on classification schemes, such as Bayesian or Nearest-Neighbors classifiers (Sebastiani, 2002; Mitchell, 1997). Practically speaking, the features of an incoming news article should be compared with the main features of (eventually related) further newsgroups. If the features match, the article could be made available to those newsgroups.
- New articles may contain emerging topics, still not covered in the current newsgroup hierarchy. Such topics should be highlighted and made available to users potentially interested in.

The problem of detecting emerging topics is quite interesting. Indeed, traditional classification schemes do not work for such an issue: since the article contains new topics, in principle no pre-existing class is suitable. For example, if topics were detected by exploiting clustering algorithms (such as, e.g., the ones devised in the previous section), no cluster could be adequate to contain the article. A naive solution can be a periodical (e.g., daily or weekly) reorganization of the clusters. Outliers can be hence collected and redistributed in the reorganization phase.

However, incremental clustering schemes can be better-suited than traditional clustering algorithms working in "batch" mode. Indeed, when new incoming documents have to be classified, a reorganization of a document collection according to a pre-existing organization can perform more efficiently than a reorganization from scratch. Moreover, updating may involve a radical restructuring only of the portion of the cluster

structure that is affected by the new document. Clearly, a reorganization involving the overall structure would be too expensive. Finally, notice that articles represent a possibly infinite stream. Thus, adopting batch restructuring could reveal unfeasible.

The problem of incrementally cluster articles can be stated as follows: given an initial cluster hierarchy $\mathcal{T}_0$ and a stream $\mathcal{S} = \{\mathcal{B}_1, \ldots, \mathcal{B}_h, \ldots, \mathcal{B}_k, \ldots, \mathcal{B}_i\}$ where $\mathcal{B}_j = \{m_{i_1}, \ldots, m_{i_{k_j}}\}$ is a collection of articles that are available at timestamp j, the objective is the computation of new hierarchies $\mathcal{T}_i$ such that *(1)* $\mathcal{T}_i$ is computed from $\mathcal{T}_{i-1}$ and $\mathcal{B}_i$, and *(2)* $\mathcal{T}_i$ is order-independent (that is, considering a different stream $\mathcal{S}' = \{\mathcal{B}_1, \ldots, \mathcal{B}_k, \ldots, \mathcal{B}_h, \ldots, \mathcal{B}_i\}$ in which two bursts hold in a different timestamp than in $\mathcal{S}$, then $\mathcal{T}_i$ is the same both in $\mathcal{S}$ and in $\mathcal{S}'$).

Several incremental clustering algorithms were proposed in the current literature. Early approaches include *COBWEB* (Fisher, 1987), originally designed for handling categorical features and based on probabilistic categorization trees, and its extension *CLASSIT* (Gennari et al., 1989) for numerical values. More recent approaches include an incremental version of *DBSCAN* (Ester et al., 1998), a density-based clustering algorithm which grows clusters according to a density threshold, and *BIRCH* (Zhang et al., 1996), which builds a tree of *clustering features*, where each clustering feature represents a cluster by means of common statistical quantities. The effectiveness of the above approaches can be strongly affected by the order in which instances are considered. That is, property *(2)* of the above formalization is not satisfied. A recent algorithm, called *GRIN* (Chen et al., 2002), seems to effectively overcome most of the above limitations.

In general, incremental clustering may benefit of ad-hoc tree-based structures, which represent clusters and summarize their key features. It is well-known that search tree indexes, such as B-trees and their variants, are particularly suitable to support dynamic insertions and deletions. A promising strategy may involve the adoption of such structures to the incremental document clustering context. For example, (Wong and Fu, 2000), adopt a B^+-tree-like structure to properly place an incoming article.

5 Mining the Usage of News Articles

This section is devoted to the problem of exploiting browsing patterns to learn user profiles. A profile is a synthetic description of the information requirements, interests and preferences of a set of visitors. Earlier approaches to personalization required explicit user collaboration, that is visitors had to fill in questionnaires concerning their navigation purposes. However, such an approach suffers from two main limitations. First, questionnaires are subjective, and hence possibly unreliable. Second, profiles learnt from questionnaires are static, i.e. they depict the expectations of a group of users at a given point in time. This eventually requires new questionnaires to be periodically presented to visitors, thus making visitors reluctant to continuously provide information.

By contrast, Web usage mining allows to infer user profiles through a silent analysis of visitors' browsing activities. Moreover, profiles learnt from browsing patterns are both objective (they are deduced from exhi-

bited navigational behaviors and therefore not affected by subjectiveness) and dynamic (they automatically reflect changes in the utilization of a Web site).

Web usage mining is at the basis of a variety of applications (Eirinaki and Vazirgiannis, 2003; Cooley, 2000) such as statistics for the activity of a Web site, business decisions, reorganization of link and/or content structure of a Web site, usability studies, traffic analysis and security. Here, we discuss how these techniques can reveal useful to the purpose of personalizing the delivery of Usenet contents.

Ideally, a typical Web usage mining process can be divided into three parts (Srivastava et al., 2000): *data preprocessing*, the process of turning raw usage data into a meaningful data set (precisely, raw usage data is converted into high-level abstractions such as page views, sessions, transactions, users); *pattern discovery*, i.e. the exploitation of a variety of techniques from different fields (such as machine learning, pattern discovery and statistics) to infer usage patterns potentially of interest; *pattern analysis*, the step in which the identified patterns are further inspected, aggregated and/or filtered to transform individual patterns into a deep understanding of the usage of the Web site under investigation. In the following, data preprocessing and pattern discovery are taken into account.

5.1 Usage data preprocessing

A number of tasks must be accomplished in order to reconstruct a meaningful high-level view of users' browsing activities from the collection of their individual actions in a Web log.

Typically, *data cleaning* is the first step. It is useful to remove irrelevant entries from Web logs, such as those whose filename suffix denotes either images, or robot accesses. The intuition is that only those entries explicitly requested by users should be retained for subsequent computations (Cooley et al., 1999): since Web usage mining aims at highlighting overall browsing patterns, it does not make sense to analyze entries which do not correspond to explicit visitor requests.

URI normalization is conceived to identify as similar those URIs which, though syntactically distinct, refer to the same resource (such as "www.mysite.com/index.htm" and "www.mysite.com").

User identification is a crucial task because of a variety of reasons such as caching policies, firewalls and proxy-servers. Many heuristics actually exist for identifying users, such as client-side tracking (i.e., the collection of usage data at client level through remote agents), cookies and embedded session IDs (a.k.a. URL rewriting), chains of references and others (Pirolli et al., 1996). It is worth noticing that in spite of a considerable number of (more or less) accurate heuristics, user identification exhibits some intrinsic difficulties, such as in the case of two users with identical IP address and user agent fields browsing the same Web pages, or in the case of an individual user who visits the same pages through two distinct Web browsers executing on a single machine.

For each user, *session identification* aims at dividing the resulting set of requests into a number of subsets (i.e. sessions). Requests in each subset share a sort of temporal continuity. It is reasonable to create a new

session for a given user, as soon as two requests from that user exceed a prefixed timeout ((Catledge and Pitkow, 1995) established that this threshold is about 25.5 minutes). Since Web logs track visitor behavior for very long time periods, session identification is leveraged to distinguish among repeated visits of a same user.

The notion of page view indicates a set of page files (such as frames, graphics and scripts) which contribute to a single browser display. *Page view identification* is then the step in which different session requests are collapsed into page views. At the end of this phase, sessions are turned into time-ordered sequences of page view accesses.

An optional preprocessing phase, *support filtering*, is typically used in order to eliminate noise from usage data, by removing all those session page views characterized by either very low or extremely high support. These accesses cannot be profitably leveraged to characterize the behavior of any group of users.

Transaction (or *episode*) *identification* is an optional preprocessing step which aims at extracting a meaningful subset of page view accesses from any user session. The notions of *auxiliary* page view (i.e., a page view mainly accessed for navigational purposes) and *media* page view (namely an informative page view) contribute to identify two main kinds of user transactions (Cooley, 2000): *auxiliary-content* and *media-only* transactions. Given a generic user session, for each media page view $\mathcal{P}$, an auxiliary-content transaction is a time ordered subsequence consisting of all the auxiliary page views leading to $\mathcal{P}$ in the original user session plus $\mathcal{P}$ itself. Browsing patterns emerging from auxiliary-content user transactions reveal the common navigation paths leading to a given media page view. Media-only transactions consist of all the media page views in a user session. They are useful to highlight the correlations among the media page views of a Web site. Four techniques are mainly exploited to classify page views into either auxiliary or media page views, namely the *page type*, *reference length*, *maximal forward reference* and *time window* methods.

5.2 Pattern discovery

A number of traditional data mining techniques can be applied to the preprocessed usage data in order to discover useful browsing patterns. Here these techniques are analyzed in the context of personalization. In the remaining part of this chapter the term page is used as a synonym for page view.

Association rules capture correlations among distinct items on the basis of their co-occurrence patterns across transactions. In a Web setting, association rule discovery can be profitably exploited to find relationships among groups of Web pages in a site (Mobasher et al., 2001). This can be accomplished through an analysis of those pages frequently visited together within user sessions. Association rules materialize the actual user judgment about the logical organization of the Web pages in a site: a group of (not necessarily inter-linked) pages often visited together implies some sort of thematic affinity among the pages themselves. Also, association rules implicitly identify sets of visitors with similar interests and preferences. *Sequential patterns* extend association rules by including the notion of time sequence. In a Web environment, a sequential

pattern indicates that a set of Web pages is chronologically accessed after another set of pages. These patterns allow Web sites to be proactive, i.e. to automatically predict current visitors' next request (Mobasher et al., 2002b). As a consequence, user navigation can be supported by suggestions to those Web pages that should best satisfy visitor browsing purposes.

Classification is a technique which assigns a given item to one of several predetermined classes. Profiling is a typical application domain: in this case, the purpose of classification is to choose, for each visitor, a (pre-existing) user profile that best reflects her/his navigational behavior (Dai et al., 2000). *Clustering* is a unsupervised technique for grouping items which exhibit similar characteristics. In a Web domain, items can be either pages or users. Page clusters consist of those Web pages thematically related according to user judgment. A technique (Mobasher et al., 2002a) for computing these clusters consists in exploiting the association rules behind the co-occurrence patterns of Web pages in such a way to form an hypergraph. Hypergraph partitioning is then applied to find groups of strongly connected pages. Different approaches (either centroid-based (Giannotti et al., 2002; Manco et al., 2003) or hierarchical (Guha et al., 2000)) are aimed at clustering users' transactions. In general, page clusters summarize similar interests of visitors with different browsing behavior. Session clusters, on the contrary, depict subsets of users who exhibit a same browsing behavior.

Statistical techniques typically exploit data within user sessions to learn a probabilistic model of the dependencies among the specific variables under investigation. In the field of personalization, an approach consists in building a Markov model (from the log data of a Web site) that predicts the Web page(s) that users will most likely visit next (Deshpande and Karypis, 2001). Markov models can also be exploited in a probabilistic framework to clustering, based on the EM (*Expectation-Maximization*) algorithm (Cadez et al., 2000). The main idea here is essentially to learn a mixture of first-order Markov models capable of predicting users' browsing behavior.

6 Mining the Structure of a Newsgroup

This section aims at investigating the basic intuitions behind some fundamental techniques in the field of Structure Mining. Though inspired to the same general principles, such techniques are however applied to a novel setting, that is the context of Usenet news articles. This allows to deal with the main concepts of structure mining without taking into account the complexities which arise in a typical Web environment.

6.1 Differences and similarities between the Web and Usenet environments

The main feature of the Web is that it can be considered as a hyperlinked media with no logical organization (Gibson et al., 1998). It results from the combination of three, independent stochastic processes of content creation/deletion/update, which evolve at various scales (Dill et al., 2001). As a consequence, even if content creators impose order on an extremely local basis (i.e., on the regions of the Web they have created),

the overall structure of the Web appears chaotic (Kleinberg, 1999). In such an environment, finding information inherent to a topic of interest often becomes a challenging task. Many research efforts in the field of structure mining are conceived to discover some sort of high-level structure to be exploited as a semantic glue for pages thematically related. The idea is that if pages with homogeneous contents exhibit some structure and if such a structure does not depend on the nature of the content, then some effective strategy can be devised in order to address the chaotic nature of the Web. Such an intuition is supported by evidence from a number of research efforts, which reveal that, from a geometric point of view, the Web can be considered as a fractal (Dill et al., 2001): highly-hyperlinked regions of the Web exhibit the same geometrical properties as the Web at large. This suggests to exploit structure mining to address relevant problems such as that of *topic distillation*, that is the identification of a set of pages highly relevant to a given topic. The basic idea is qualitatively the following. Given a topic of interest, the focus of the approach is on the search for a known link structure surrounding the pages relevant to that topic. This requires exploiting only an extremely limited portion of the Web as a starting point. However, since hyperlinks encode a considerable amount of latent human judgment (Kleinberg, 1999), the structure among the initial pages allows the algorithm to collect most of the remaining relevant pages: what the algorithm needs to do is, in a certain sense, choosing and following the structural connections (leading from the initial pages to unexplored regions of the Web) which are indicative of contents as prominent as those within the original pages.

Traditional approaches to information discovery on the Web are, on the contrary, negatively affected by the chaotic nature of the Web itself. This characteristic seems to make fruitless every attempt at devising an endogenous measure (of the generic Web page) to assess the relevance of a given page to a certain subject. This is why the search for relevant pages through search engines is often frustrating. In fact, all the pages in a response to given query must be investigated in order to find those that are really of interest. Moreover, such an approach inevitably requires to deal with two more issues, which are typical of a Web environment: the *scarcity problem*, which arises when specific queries narrow the search space to only a very few pages scattered on the Web, and the *abundance problem*, which on the contrary is determined by broad-topic queries involving a huge number of pages (Kleinberg, 1999).

A newsgroup environment (such as that of Usenet) benefits from a number of interesting features which contribute to make it an ideal application domain for structure mining techniques. Usenet overall structure, in fact, appears as a network of news articles, semantically organized by topic. This is mainly due to fact that users typically post news articles pertaining to a specific (more or less broad) topic. Therefore, every single user is involved to play a crucial role in the process of keeping the overall structure of any newsgroup ordered: the destination group of a news article is already established since the early stages of the inception of the article itself. Two more elements contribute to impose order over the entire structure of a newsgroup. First, the evolution of any newsgroup is based on the sole process of news article posting: no content deletion and/or update is allowed. Second, in contrast with what happens on the Web, the process of content management

is centralized: many users post their own articles to the same recipient entity, which is the only responsible of their availability through the Web.

Newsgroups and the Web present many significative differences. However, some similarities can be highlighted. Newsgroups are available from the Web and this implies that every news article is a Web page, characterized by a proper content and its own structure. Users browse through the news articles within each group in order to find those which are mostly of interest: this is similar to what happens on the Web, though considerable differences characterize the search space in the two cases. Article browsing is made possible by a link topology that exists within each group and among the news articles belonging to distinct groups. Such a structure consists of three different kinds of links: *group links*, which belong to the hierarchy of newsgroups and connect a news article to its own group; *explicit links*, which correspond to links within the content of an article; and *implicit links*, which can be devised in order to model a number of distinct correlations among the news articles (for instance, implicit links can depict article dependencies such as the reply-to relation).

The peculiarities of newsgroups as well as their affinities with the Web are at the basis of our intuition: applying known structure mining techniques to Usenet. If approaches based on link analysis exhibit good performances on the Web, their behavior should be at least as interesting if applied to a newsgroup environment. In fact, in the former case link structure is implicit, that is it needs to be reconstructed from huge collections of Web pages, in the latter case it is explicitly available (articles are already grouped on the basis of the homogeneity of their topics). Moreover, while the Web requires dealing mainly with *inter-site* link structure, Usenet only requires processing the *intra-site* structure of the newsgroup.

6.2 HITS algorithm

In (Kleinberg, 1999) Kleinberg introduces HITS, an effective approach for discovering Web pages relevant to a particular topic of interest. Precisely, Kleinberg distinguishes between *authoritative* pages, i.e. pages conveying prominent information, and *hub* pages, which consist of collections of links pointing to the authorities. The intuition is that if a page is an authority, then there should be a considerable number of hub pages (scattered on the Web) pointing to that authority. Intuitively, a mutually reinforcing relationship keeps authoritative and hub pages together: a good hub links to many good authorities, while a good authority is linked to by many good hubs. As a consequence, given a specific topic, hub pages play a crucial role in the process of identifying relevant information: since they should point to nearly all the authorities on that topic, hubs can be considered as a sort of glue which keeps authorities together. The notion of hubs and authorities and the associated mutually reinforcing relationship materialize in a precise link structure, which is independent of the topic under investigation. Figure 5 (a) illustrates the geometrical structure lying behind the notion of hubs and authorities.

HITS can be summarized as follows. The algorithm takes as input a focused subgraph $\mathcal{G}$ of the Web (in which nodes correspond to pages and edges to hyperlinks among pages), and computes hubs and authorities

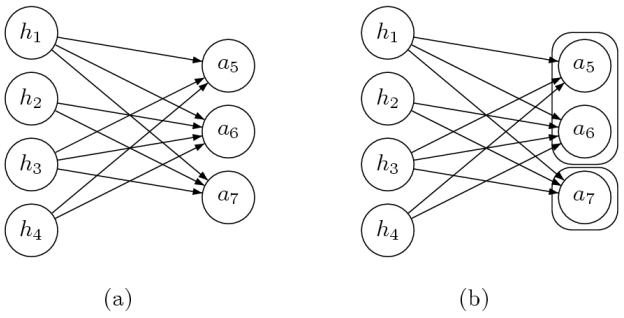

Fig. 5. The structural patterns behind hubs and authorities

on the basis of the intuition above discussed. Precisely, authority and hub weights are assigned to each page p in $\mathcal{G}$: respectively x_p and y_p. A limited number of iterations is required before the algorithm converges. Each iteration consists of two steps. First, authority and hub weights are updated for each page in $\mathcal{G}$. Then, both kinds of weights are normalized. Formally, given a page p, its associated weights are updated on the basis of the operations below:

$$x_p = \sum_{q:(q,p)\in\mathcal{G}} y_q \qquad\qquad y_p = \sum_{q:(p,q)\in\mathcal{G}} x_q$$

At the end of the third phase, authorities (resp. hubs) correspond to the k pages with highest authority (resp. hub) weight.

6.3 Bringing hubs and authorities to the surface in the Usenet environment

A link structure can be exploited in order to infer a notion of *community*, i.e. a set of pages which exhibit a sort of thematic cohesion. This is due to the fact that a high density of linkage from hubs to authorities turns into a thematic cohesion of the involved entities. Since content homogeneity seems to depend mainly on topological properties, we claim that structure mining techniques can be profitably applied even in the restricted context of a newsgroup, provided that the latent structure among news articles is suitably reconstructed. A number of application scenarios are discussed next.

Finding authoritative articles. The problem of locating articles relevant to a topic of interest can be addressed in a way which is slightly different from the original intuition behind HITS. A root set is formed by choosing (nearly) all those groups which are inherent to the specific topic. The root set is then extended including any news article within each of the above groups: given the limited size of news articles, there should be no need to sample the groups under investigation. In such a scenario, both explicit and implicit links among news articles are exploited to model the conferral of authority from hubs to authorities. For instance, consider a discussion group on a generic topic $\mathcal{T}$. A user u_i could post an article m_i containing questions about $\mathcal{T}$.

Another user, u_j, could reply to u_i by posting an article m_j with the required answers. m_i and m_j are tied by a reply-to relation: (m_i, m_j) is an implicit, directed link of the graph $\mathcal{G}$ which, in turn, represents the overall network of dependencies among the news articles in the base set. Also, the content of m_j could include hyperlinks referring to other news articles. For any such link pointing to a destination article m_k, an explicit directed link (m_j, m_k) is added to $\mathcal{G}$. An in depth analysis of the resulting network of dependencies can highlight interesting (and often unexpected) structural patterns. Two interesting situations are discussed below.

First, a community dedicated to a given topic could include hubs from different groups, all referring to authorities in a same group. This could be due to that fraction of users who send their questioning articles to groups which are not strictly related to the article contents. Typically, replies from more experienced users could refer the strongest authorities in the group which is closest to the contents of the above news articles.

Second, authorities in distinct groups could be pulled together by hubs in a same group. This could happen with topics having different meanings. If one of these meaning is overly popular with respect to the others, then its corresponding group is likely to collect even those articles concerning less popular facets of the general topic. Again, replies could show the proper authorities for such news articles.

Finding authoritative sites: articles as hubs. This application aims at discovering hybrid communities, i.e. highly-cohesive sets of entities, where hubs correspond to news articles and authorities to Web sites.

Given a topic of interest, a root set is constructed as in the above case. However, only explicit links are taken into account during the process of reconstructing the link structure of the newsgroup region under investigation. The linkage patterns behind hybrid communities show strong correlations between Usenet and the Web: precisely, contents on the latter either complement or detail information within the former.

The geometrical characterization of the notion of hubs and authorities in terms of a bipartite graph structure in Figure 5 (a) also applies to the Usenet environment, as far as the above two applications are concerned.

Authoritative people: articles as hubs. The network of implicit links among articles is rich in information useful for finding authoritative people. Conceptually, since replies determine the introduction of implicit links into the network of news articles, any such link is indicative of an endorsement of the authority of the associated replying author.

Precisely, an author who answers to a considerable amount of user questions on a certain theme can be considered as an authority on that topic. Authoritative people emerge from an in-depth analysis of heterogeneous, bipartite graph structures which can be located within the overall article network. Heterogeneity depends on the intrinsic nature of these structures, where an hub is a questioning article and an authority, instead, corresponds to a collection of replies grouped by their author (substantially, an authority is a set of nodes in the original article graph collapsed into a single entity). Figure 5 (b) shows the geometrical characterization of the notion of hubs and authorities in this context. Authoritative people can be discovered

at different extents: either for any subset of specified topics in the newsgroup, or taking into account all of the topics in the newsgroup itself. Both cases are useful to highlight interesting structural patterns which are not obvious at a first sight, such as those individuals who are authorities on a number of either related or unrelated topics.

7 Providing Personalized Access to Usenet Communities

Personalization can be defined as the process of tailoring the delivery of contents (or services) in a Web site to address individual users' features such as their requirements, preferences, expectations, background and knowledge. It is a wide research and industrial area which comprises notions such as *recommender systems* and *adaptive Web sites*. Recommender systems are conceived for the automatic delivery of suggestions to visitors: suggested items can be information, commercial products or services. By contrast, adaptive Web sites address visitors' features mainly by means of two adaptation methods (Brusilovsky, 1996): *adaptive presentation*, namely adapting the contents in a Web page to the profile of the browsing visitor, and *adaptive navigation support*, that is suggesting the right navigation path to each individual user on the basis of her/his browsing purposes.

From a functional point of view, Web sites with personalization capabilities can be designed around the notions of either *customization* or *optimization* (Perkowitz and Etzioni, 1998). Customization is adapting the content delivery of a Web site to reflect the specific features of individual users: precisely, changes to the site organization (i.e., to both contents and structure) are brought on the basis of the features of each single visitor. Optimization, on the contrary, is conceived to modify the site structure in order to make the site itself easier to use for all visitors, including those who have never visited it before.

Changes that can be brought to the structure of a Web site can be of two kinds: either strategic or tactical changes. The former type implies a modification of the link structure of the Web site (for instance when new links are added to the pages), while the latter consists of soft changes such as highlighting the page links potentially of interest to the visitors. Personalization strategies based on tactical changes can be adopted when strategic changes seem to either confuse visitors or result in a loss of those benefits deriving from the original design of the Web site (Coenen et al., 2000).

As far as the technologies behind personalization systems, four main categories can be identified (Mobasher et al., 2000): decision rules, content-based filtering, collaborative filtering and Web usage mining.

Decision rules allow to explicitly specify how the process of content delivery can be affected by the profile of visitors. Such rules can be either inferred from user interactions with the Web site or manually specified by a site administrator. Systems based on content-filtering (Lieberman, 1995) learn a model of user interests in the contents of the site by observing user navigation activities for a period of time. Then, they try to estimate visitors' interest in documents not yet viewed on the basis of some similarity between these documents and the profile to which visitors themselves belong to. Collaborative filtering systems explicitly

ask for users' ratings or preferences. A correlation technique, such as Nearest-Neighbors, is then leveraged to match current user's data with the information already collected from previous users. A set of visitors with similar ratings is chosen and, finally, suggestions that are predicted to best adapt to current users' requirements (Konstan et al., 1997) are automatically returned.

Recently, an increasing focus has been addressed to techniques for pattern discovery from usage data. Not only does Web usage mining prove to be a fertile area for entirely new personalization techniques, but it also allows to improve the overall performances of traditional approaches. For example, content-based personalization systems may fail at capturing latent correlations among distinct items in a Web site. Such a limitation can be avoided by taking into account evidence from user browsing behavior. Also, collaborative filtering can be revised in order to avoid the drawbacks due to the collection of static profiles. Precisely, users are no more correlated on the basis of the information provided by themselves. Rather, they are compared in terms of similarities among their browsing sessions.

Next we discuss how Web usage mining techniques can be applied to the problem of providing a personalized access to Usenet communities. To this purpose, we first introduce PageGather (Perkowitz and Etzioni, 1998), a technique conceived for generic Web sites. Then we show how it can be partially modified to be exploited in a Usenet environment.

7.1 PageGather algorithm

PageGather is an optimization approach which achieves the goal of supporting user navigation through a given Web site. Precisely, indexes for the pages in a Web site are automatically created: each index references a group of Web pages perceived by visitors as related. This allows visitors to directly access all the pages inherent to a given topic of interest, without having to locate them within the link structure of the Web site. A visit-coherence assumption is at the basis of the algorithm: the pages visited by an individual user, during a navigation session, tend to be conceptually related. The algorithm can be divided into three main steps.

– *Evaluation of similarities among Web pages.* For each pair of pages p_i and p_j in the Web site, both the probabilities $\Pr(p_i|p_j)$ and $\Pr(p_j|p_i)$ of a user accessing a page p_i (resp. p_j), provided that she/he visited p_j (resp. p_i) in the same session, are computed. Then, the co-occurrence frequency between p_i and p_j is chosen to be the minimum of the two probabilities. Such a choice allows to avoid mistaking asymmetrical relationships for true cases of similarity. For instance, if p_i is a popular page then $\Pr(p_i|p_j)$ may be high. Such an evidence would imply that p_i and p_j are strongly correlated. However, $\Pr(p_j|p_i)$ may be low and such an evidence should be taken into account before establishing a correlation degree between the two pages. A similarity matrix is then formed, where the (i, j)-th entry is the co-occurrence frequency between p_i and p_j if there is no link between the two pages, otherwise the entry is 0. In order to reduce the effect of noise, all entries with values below a given threshold are set to 0.

– *Cluster mining.* A graph is formed from the similarity matrix. Here nodes correspond to Web pages and edges to the nonzero entries of the matrix. Two alternatives are now possible: finding either cliques (i.e., a group of nodes such that every pair of nodes is linked by an edge) or connected components (i.e., a set of nodes, where every pair of nodes is connected by a path). The former approach allows to discover highly cohesive clusters of thematically related pages. The latter is computationally faster and leads to the discovery of clusters made up of a higher number of pages.
– *Automatic creation of indexes for each group of related pages.* For each discovered cluster, an index consisting of links to every Web page in that cluster is generated. Indexes become preferred entry points for the Web site: visitors only have to choose an index dealing with the topic of interest.

Usenet consists of a set of newsgroups hierarchically classified by topic. As a consequence, the process of finding interesting news articles necessarily implies a two phase search. First, any newsgroup which deals with the required topic have to be identified. Second, either trivial or uninteresting threads within each such newsgroup need to be filtered out. Such a laborious task may disorientate users, not only those who have never accessed Usenet before, but also experienced visitors whose search activities may be biased by the huge number of both newsgroups and threads.

We propose an approach for personalizing user access to Usenet news articles based on the exploitation of PageGather. The idea consists in providing users with a new logical organization of the threads, which captures their actual perception about the inter-thread correlations. Here, the visit-coherence assumption is made w.r.t. threads. Statistical evidence, mediated on a huge number of browsing patterns, allows to find clusters of related threads, according to the visitors' perception of the logical correlations among the threads themselves. Each such cluster deals with all the facets of a specific topic: it represents an overall view of the requirements, preferences and expectations of a subset of visitors. Finally, an index page could be associated to each individual cluster in order to summarize its contents and quickly access to the specific facet of the corresponding topic.

Our proposal benefits of two main advantages. First, a more effective categorization of Usenet contents is provided to visitors. Second, the technique is an optimization approach to the delivery of Usenet contents: it does not require that strategic changes are brought to the original structure of Usenet newsgroups.

8 Conclusions

We analyzed three main lines of investigation, which may contribute in providing personalized access to Usenet articles available via a Web-based interface.

Current content mining techniques were applied on the management of news articles. To this purpose, we mainly studied the problem of classifying articles according their contents. We also highlighted the streaming nature of Usenet contents, and provided an overview of incremental methods to accomplish incremental mining.

We provided an insight of techniques for tracking and profiling users, in order to infer knowledge about their preferences and requirements. Our interest focused on techniques and tools for the analysis of application/server logs, comprising data cleaning and preprocessing techniques for log data, and log mining techniques.

Also, we modeled the events which typically happen in a Usenet scenario adopting a graph-based model, analyzing the most prominent structure mining techniques, such as discovery of hubs and authorities, and discovery of Web communities. We finally analyzed personalization methodologies, mainly divided into optimization and customization approaches, that can benefit from a synthesis of the above described techniques.

It is important to stress here that, from the viewpoint of the devised application, all the techniques studied in this paper can influence each other. The strategies for tracking and profiling users' behavior should widely benefit from the overall organization of articles available in the newsgroups, which ultimately relies from the content mining techniques adopted. A dependency relationship also relates personalization and profiling to a combination of structure and usage mining techniques suitably conceived for the Usenet environment. These considerations make the devised application scenario particularly significant in the context of Web Mining and personalization.

Bibliography

[Agrawal and Srikant, 1994]Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. Very Large Data Bases, (VLDB'94)*, pages 487–499.

[Baeza-Yates and Ribeiro-Neto, 1999]Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. ACM Press Books. Addison Wesley.

[Beil et al., 2002]Beil, F., Ester, M., and Xu, X. (2002). Frequent term-based text clustering. In *Proc. 8th ACM Conf. on Knowledge Discovery and Data Mining (KDD'02)*, pages 436–442.

[Brusilovsky, 1996]Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction*, 6(2-3):87–129.

[Cadez et al., 2000]Cadez, I., Gaffney, S., and Smyth, P. (2000). A general probabilistic framework for clustering individuals and objects. In *Proc. 6th ACM Conf. on Knowledge Discovery and Data Mining (KDD'00)*, pages 140–149.

[Catledge and Pitkow, 1995]Catledge, L. and Pitkow, J. (1995). Characterizing browsing behaviors on the world wide web. *Computer Networks and ISDN Systems*, 27(6):1065–1073.

[Chakrabarti, 2002]Chakrabarti, S. (2002). *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan-Kaufmann.

[Chen et al., 2002]Chen, C., Hwang, S., and Oyang, Y. (2002). An incremental hierarchical data clustering algorithm based on gravity theory. In *Proc. 6th Pacific-Asia Conf. (PAKDD'02)*, pages 237–251.

[Coenen et al., 2000]Coenen, F., Swinnen, G., Vanhof, K., and Wets, G. (2000). A framework for self adaptive websites. In *Proc. ACM SIGKDD Workshop on Web Mining for e-commerce - Challenges and opportunities (WEBKDD'00)*.

[Cooley, 2000]Cooley, R. (2000). *Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data*. PhD thesis, University of Minnesota.

[Cooley et al., 1999]Cooley, R., Mobasher, B., and Srivastava, J. (1999). Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1(1):5–32.

[Dai et al., 2000]Dai, H., Luo, T., Mobasher, B., Sung, Y., and Zhu, J. (2000). Integrating web usage and content mining for more effective personalization. In *Proc. Int. Conf. on E-Commerce and Web Technologies (ECWeb'00)*, volume 1875 of *Lecture note in Computer Science*, pages 165–176.

[Deshpande and Karypis, 2001]Deshpande, M. and Karypis, G. (2001). Selective markov models for predicting web-page accesses. In *Proc. SIAM Int. Conf. on Data Mining (SDM'01)*.

[Dhillon and Modha, 2001]Dhillon, I. and Modha, D. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1/2):143–175.

[Dhyani et al., 2002]Dhyani, D., Ng, W., and Bhowmick, S. (2002). A survey of web metrics. *ACM Computing Surveys*, 34(4):469–503.

[Dill et al., 2001]Dill, S., Kumar, S., McCurley, K., Rajagopalan, S., Sivakumar, D., and Tomkins, A. (2001). Self-similarity in the web. In *The VLDB Journal*, pages 69–78.

[Eirinaki and Vazirgiannis, 2003]Eirinaki, M. and Vazirgiannis, M. (2003). Web mining for personalization. *ACM Transactions on Internet Technology*, 3(1):1–27.

[Ester et al., 1998]Ester, M., Kriegel, H., Sander, J., Wimmer, M., and Xu, X. (1998). Incremental clustering for mining in a data warehousing environment. In *Proc. 24th Int. Conf. on Very Large Data Bases (VLDB'98)*, pages 323–333.

[Fisher, 1987]Fisher, D. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172.

[Gennari et al., 1989]Gennari, J., Langley, P., and Fisher, D. (1989). Models of incremental concept formation. *Artificial Intelligence*, 40:11–61.

[Giannotti et al., 2002]Giannotti, F., Gozzi, C., and Manco, G. (2002). Clustering transactional data. In *Proc. 6th European Conf. on Principles and Practices of Knowledge Discovery in Databases (PKDD'02)*, pages 175–187.

[Gibson et al., 1998]Gibson, D., Kleinberg, J., and Raghavan, P. (1998). Inferring web communities from link topology. In *Proc. 9th ACM Conf. on Hypertext and Hypermedia*, pages 225–234.

[Guha et al., 2000]Guha, S., Rastogi, R., and Shim, K. (2000). ROCK: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345–366.

[Huang, 1998]Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304.

[Jain et al., 1999]Jain, A., Murty, M., and Flynn, P. (1999). Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323.

[Kleinberg, 1999]Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

[Konstan et al., 1997]Konstan, J. et al. (1997). Grouplens: applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87.

[Kosala and Blockeel, 2000]Kosala, R. and Blockeel, H. (2000). Web mining research: A survey. *SIGKDD Explorations*, 2(1):1–15.

[Lieberman, 1995]Lieberman, H. (1995). Letizia: An agent that assists web browsing. In *Proc. 14th Int. Joint Conf. on Artificial Intelligence (IJCAI'95)*, pages 924–929.

[Manco et al., 2002]Manco, G., Masciari, E., and Tagarelli, A. (2002). A framework for adaptive mail classification. In *Proc. 14th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI'02)*, pages 387–392.

[Manco et al., 2003]Manco, G., Ortale, R., and Saccà, D. (2003). Similarity-based clustering of web transactions. In *Proc. ACM Symposium on Applied Computing (SAC'03)*, pages 1212–1216.

[Mitchell, 1997]Mitchell, T. (1997). *Machine Learning*. Computer Sciences Series. McGraw-Hill.

[Mobasher et al., 2000]Mobasher, B., Cooley, R., and Srivastava, J. (2000). Automatic personalization based on web usage mining. *Communications of the ACM*, 43:142–151.

[Mobasher et al., 2001]Mobasher, B., Dai, H., Luo, T., and Nakagawa, M. (2001). Effective personalization based on association rule discovery from web usage data. In *Proc. 3rd Int. Workshop on Web Information and Data Management (WIDM'01)*, pages 9–15.

[Mobasher et al., 2002a]Mobasher, B., Dai, H., Luo, T., and Nakagawa, M. (2002a). Discovery and evaluation of aggregate usage profiles for web personalization. *Data Mining and Knowledge Discovery*, 6(1):61–82.

[Mobasher et al., 2002b]Mobasher, B., Dai, H., Luo, T., and Nakagawa, M. (2002b). Using sequential and non-sequential patterns for predictive web usage mining tasks. In *Proc. IEEE Int. Conf. on Data Mining (ICDM'02)*, pages 669–672.

[Moens, 2000]Moens, M. (2000). *Automatic Indexing and Abstracting of Document Texts*. Kluwer Academic Publishers.

[Perkowitz and Etzioni, 1998]Perkowitz, M. and Etzioni, O. (1998). Adaptive web sites: Automatically synthesizing web pages. In *Proc. 15th Nat. Conf. on Artificial Intelligence (AAAI'98)*, pages 727–732.

[Perkowitz and Etzioni, 2000]Perkowitz, M. and Etzioni, O. (2000). Towards adaptive web sites: Conceptual framework and case study. *Artificial Intelligence*, 118(1-2):245–275.

[Pirolli et al., 1996]Pirolli, P., Pitkow, J., and Rao, R. (1996). Silk from a sow's ear: Extracting usable structures from the web. In *Proc. ACM Conf. Human Factors in Computing Systems (CHI'96)*, pages 118–125.

[Sebastiani, 2002]Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

[Srivastava et al., 2000]Srivastava, J., Cooley, R., Deshpande, M., and Tan, P. (2000). Web usage mining: Discovery and applications of web usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23.

[Steinbach et al., 2000]Steinbach, M., Karypis, G., and Kumar, V. (2000). A comparison of document clustering techniques. In *Proc. ACM SIGKDD Workshop on Text Mining*.

[Strehl et al., 2000]Strehl, A., Ghosh, J., and Mooney, R. (2000). Impact of similarity measures on web-page clustering. In *Proc. AAAI Workshop on Artificial Intelligence for Web Search*, pages 58–64.

[Wong and Fu, 2000]Wong, W. and Fu, A. (2000). Incremental document clustering for web page classification. In *IEEE Int. Conf. on Information Society in the 21st Century (IS'00)*.

[Zhang et al., 1996]Zhang, T., Ramakrishnan, R., and Livny, M. (1996). BIRCH: An efficient data clustering method for very large databases. In *Proc. ACM Conf. on Management of Data (SIGMOD'96)*, pages 103–114.