User recommendation for collaborative and personalised digital archives

Henri Avancini and Umberto Straccia*

Istituto di Scienza e Tecnologie dell'Informazione – C.N.R.,
Via G. Moruzzi, Pisa 1 I-56124, Italy
Fax: +39 050 315 3464 E-mail: avancini@isti.cnr.it
E-mail: straccia@isti.cnr.it
*Corresponding author

Abstract: Every day, a huge amount of newly created information is electronically published in digital libraries. Complementary to the usual vision, we envisage a digital library not only as an information source where users may submit queries to satisfy their daily information need, but also as a collaborative working and meeting space of people sharing common interests.

Keywords: digital library; collaboration; personalisation; recommendation.

Reference to this paper should be made as follows: Avancini, H. and Straccia, U. (2005) 'User recommendation for collaborative and personalised digital archives', *Int. J. Web Based Communities*, Vol. 1, No. 2, pp.163–175.

Biographical notes: Henri Avancini is a PhD student at the Istituto di Scienze e di Tecnologie dell'Informazione (ISTI) of the Italian National Council of Research (CNR). His main interests include machine learning (ML) techniques, software engineering, multi-agent systems (MAS), information retrieval (IR) and text categorisation (TC), with particular regard to hierarchical text categorisation (HTC). He developed an MAS framework and work on filtering and recommender systems. He is at present working on digital libraries and grid technologies.

Umberto Straccia is researcher at the Istituto di Scienze e di Tecnologie dell'Informazione (ISTI) of the Italian National Council of Research (CNR). His main research interests include in the broad sense information retrieval (IR) and knowledge representation and reasoning (KRR). In particular, he has interests in logics for KRR, user profiling, and recommender systems. The activities have been mainly carried out in the context of EU funded projects he conducted or has been involved in.

1 Introduction

A characteristic of the digital information age is that the amount of information published in electronic format, the services provided on it and the number of users accessing it to satisfy their daily information need is growing at a tremendous rate. In this scenario, digital libraries (DLs) (Fox and Marchionini, 2001) will play an important role in the near future not merely in terms of the 'controlled' digital information (the content of DLs) they allow access to, but especially in terms of the services they provide. Informally, DLs

can be defined as consisting of collections of information, which have associated services delivered to users and user communities using a variety of technologies. The information accessible from DLs is usually heterogeneous both in content (addressing many aspects of human knowledge) and format and can be represented as digital text, image, audio, video, or other media. This information can be digitised paper or from digital material. The services offered on such information can be varied, ranging from content operations to rights management and can be offered to individuals as well as to user communities. Indeed, an essential technology component of DLs is that they are networked, meaning that access is increasingly becoming shared and collaborative.

Without doubt, DLs have evolved rapidly over the past decade and as DLs become more commonplace and the range of information they provide services increases, users are expecting more and more sophisticated services from their DLs. In fact, more and more DLs, rather than providing a search facility to the digital society at large (they are oriented towards a generic user, as they answer queries crudely rather than learn the long-term requirements of a specific user), are going to move from being passive with little adaptation to their users, to being more proactive and personalised in offering and tailoring information for individual users. For instance, towards this direction fall those DLs, which offer the so-called personalised alerting services, see for e.g., Bollacker et al. (1999), i.e., services that notify a user (usually, by sending an e-mail), with a list of references to newly available documents in the DLs and deemed as relevant to some of the (manually) user specified topics of interests. Some other DLs, in addition, support the users in being able to organise their information space they are accessing to according to their own subjective perspective, see for e.g., Fernandez et al. (2000). In particular, they usually allow to organise the data retrieved in a DL into thematic folders, like computer users do within their own computer. This is important as users and communities of users might well profit from being able to organise the information space in a personalised fashion both in terms of restricting the information space in which to search into as well as in terms of organising it not necessarily in the way the DL manager thought would be well suited for anyone.

Our vision is that, DLs can also be considered as collaborative meeting place of people sharing common interests. Indeed, DLs may be viewed as a common working place where users may become aware of each other (indeed a DL may find out interesting relationships both between users and/or between communities of users and produce the appropriate recommendations/advices), open communication channels and exchange information and knowledge with each other or with experts. In fact, usually users and/or communities access a DL in search of some information. This means that it is quite possible that users may have overlapping interests if the information available in a DL matches their expectations, backgrounds, or motivations. Such users might well profit from each other's knowledge by sharing opinions or experiences or offering advice. Some users might enter into long-term relationships and eventually evolve into a community if only they were to become aware of each other. Such a service might be important for a DL as it supplies very focused information. Hence, we are moving from services supporting an individual user towards services supporting groups (or a community) of users: thus, move from the study of individual human behaviour towards the discipline concerned with the study of human behaviour in groups and the technical support thereof. More fundamentally, we make a conceptual shift in our understanding of DLs: whereas the classical view of DLs was manipulation of data by isolated individuals, our view of DLs is manipulation and exchange of data and

information as well as cooperation by individuals aware of their environment as well as other users. We have developed a system, named CYCLADES (http://www.ercim.org/cyclades), whose aim is to provide advanced services for both personalisation and collaborative work. The description of its features is one of the topics of this paper. Besides, we present some experimental results showing the effectiveness of CYCLADES in relating users with similar interests to each other. This is important, as described above, as it may help users to become aware of other users sharing similar interest and to enter into long-term relationships. These users may grow-up ultimately to a community.

The outline of the paper is as follows. In the next section we recall the main features of CYCLADES, while in Section 3 we report some experimental results of the user recommendation algorithms adopted within CYCLADES. Section 4 concludes the paper.

2 CYCLADES

The CYCLADES system offers a broad range of functionality for both individual scholars, who wish to search and browse in digital archives, as well as for communities of scholars who wish to share and exchange information. A multi-disciplinary and distributed team designed this functionality with backgrounds in digital libraries, databases, information retrieval, web-based systems, as well as computer-supported cooperative work and virtual communities. After a first suggestion for the basic functionality of the system was completed, this functionality was presented to users and their feedback was captured via a web-based questionnaire.

The CYCLADES system integrates a set of functionalities that support the user when accessing very large virtual e-print archives with: functionality for efficient and effective retrieval of relevant information from many large, distributed and multi-disciplinary digital archives; feedback on the degree of relevance of the retrieved information; regular information about new publications in the archive environment that is relevant to the users' interests; automatic retrieval of users' long-term information needs; as well as rapid dissemination of the search results world-wide. A special set of features provides communities of scholars with functionality for: the dissemination of relevant information to community members in the form of recommendations, which are based on collective profiles and behaviour; very quick online annotations on research results published by members of the community; carrying out community services such as peer review, which requires the annotation of online papers by reviewers and the sharing of these annotations among editors, authors and others; as well as enabling community members to learn from, contribute to and collectively build upon the community's knowledge.

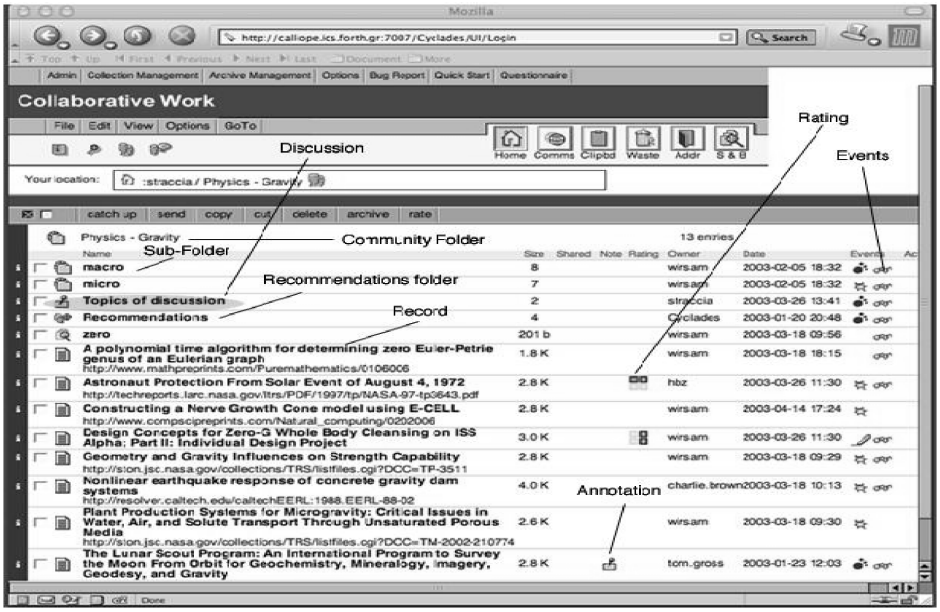
The digital archives to which CYCLADES users have access to are those adhering the open archives initiative (OAI) (http://www.openarchives.org). Informally, the OAI is an agreement between several digital archive providers in order to provide some minimal level of interoperability between them. In particular, the OAI defines an easy-to-implement gathering protocol over HTTP, which give data providers (the individual archives) the possibility to make the documents' metadata in their archives externally available. Indeed, the agreement specifies that each document of an archive should posses a metadata record describing the documents properties and content. In particular, the format of the metadata records should be Dublin Core (http://dublincore.org). The metadata record consists of several attributes describing author, title, abstract, etc., of a document. The protocol allows then to gather these

metadata records (in place of the real documents). A link to the 'real' document is also present if the document is accessible. A metadata record may be understood as a statement of existence and short description of a document, which may be then accessible to a user according to the access policies of the archive, which owns the document. To date, there is a wide range of archives available (more than 100 registered archives) in terms of its content, forming a quite heterogeneous and multi-disciplinary information space. In CYCLADES we gather these records periodically from the archives.

In order to use the CYCLADES environment and its functionality users have to register with the system. As the user interface is web-based, users simply enter a valid e-mail address in the registration web page. After the user has carried out these steps, the user is registered and has a login name, password and (empty) home folder. Also, CYCLADES provides a folder-based environment (Figure 1 shows the content of a user folder, in our case the 'physics-gravity' folder of the community of physicists) for managing, e.g., metadata records, queries, external documents, ratings and annotations. In particular, users may organise their own information space according to their own hierarchy of folders. Each folder typically corresponds to one user related subject, discipline, or field, so that it may be viewed as a thematic and usually semantically related repository of data items. This environment is an extension of the basic support for collaborative work environment (BSCW) (http://www.bscw.de, it is not accidental that the BSCW development team was member of the project); see Bentley et al. (1997). Users can then login in order to access their home folder or add some additional user information such as full name, affiliation, postal address, phone and fax numbers and so forth. Furthermore, users can start working with folders and folder contents. They can create private folders, community folders, as well as project folders and add subfolders and documents to them. Private folders and community folders can contain metadata records and queries; project folders additionally can contain any type of document (e.g., PDF-files, slides, text files). The contents of private folders can be seen, accessed and changed only by its owner. The contents of community and project folders can be seen, accessed and changed by the community and project members, respectively. Users can add documents in various ways: they can store records they found in a search and browse activity, they can upload documents from the local hard disks, they can add links and so forth. Community folders may also contain discussion forums where notes may be exchanged in threaded discussions (similar to news groups). All documents can be rated on a simple scale. The median of the ratings is then attached to the document and visible for any user who has access to the respective document. Additionally, annotations containing free text can be added to documents. Annotations can also be seen by any user who has access to the respective document.

In order to become member of an existing community users have two options: either they browse a list of names and descriptions of open communities and subscribe to them according to their interests or they get invited by community managers of closed communities. After a user has become a member, the community folder is visible and accessible from the user's home folder. If users leave a community, the community folder is removed from their home folder, but is still available for the other community members. Only when the last community member leaves the community, the community folder is actually deleted.

Figure 1 User interface: folder document



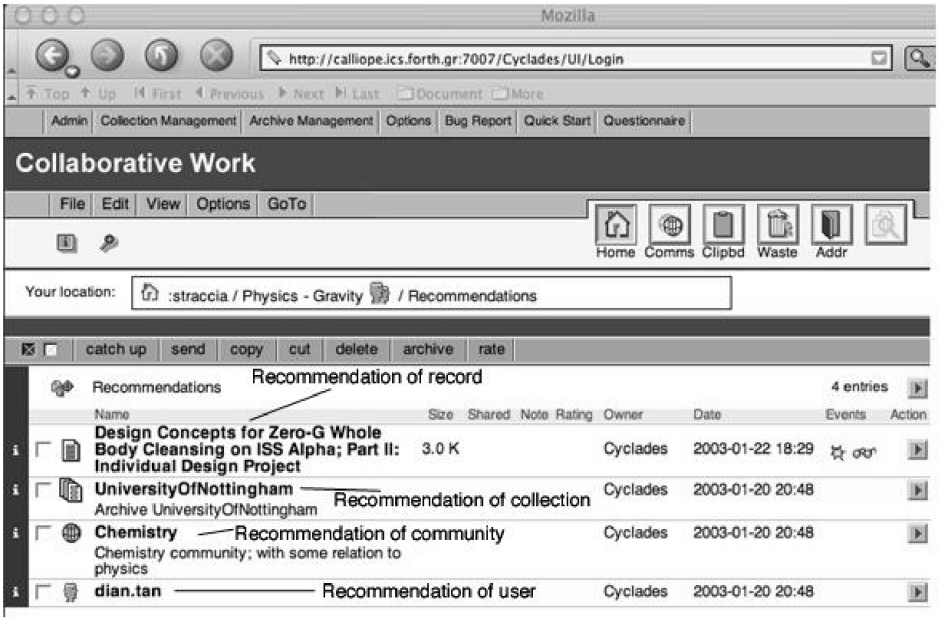
For not to lose shared activity in the collaborative DL environment, mutual awareness is supported. Indeed, the CYCLADES environment provides two features for informing users about activities in the system: event icons and activity reports. Event icons are attached to the individual documents and show recent actions that were performed on the documents (e.g., creation, change, read). Activity reports are sent out daily via e-mail; they contain information about changes to documents since the last report was sent out. Besides the possibility to search into the folders a user has access to, advanced functionality of searching records in the various collections accessible from within CYCLADES is provided. Users can issue a query and are allowed to store selected records within the folders and community folders they have access to. Essentially, three types of search are supported:

- In ad-hoc search a user specifies a query and the system looks for relevant records within specified collections.
- Filtered search is like the usual ad-hoc search, except that the user specifies, additionally to a query (e.g. ‘zero’), also a target folder (e.g., ‘physics-gravity’). The goal of the system is then to find documents not only relevant to the query, but also those to the topic of the target folder (in our example, the request is something like ‘find records about zero gravity’)
- In what’s new, on-demand, the user specifies one of his folders, without specifying a query and the goal of the system consists in finding all records, relevant to the folder, which would become available to the system since the last time the user asked for this request. This corresponds roughly to the functionality provided by alerting services, except that the topic a folder is about is built automatically from the folder content (we call the topic of a folder, folder profile) and that records are delivered to the user on-demand.

An important service in CYCLADES is the automatic delivering of recommendations to a user. The key point is that all recommendations are specific to a given user folder (topic of interest), i.e., they have always to be understood in the context not of the general interests of the user, but of the specific interests (topic) of the user represented by a folder. For instance, Figure 2 shows the recommendations related to the 'physics-gravity' folder, deemed by the system as relevant to this folder. The user Dian is considered by the system to have overlapping interests with 'Physics' and 'Gravity'. Also, CYCLADES provides types of recommendations. A user folder may get recommendations of

- collections, suggesting to the user that recommended collections contain relevant data with respect to the folder and thus, it may be worth while to search within them
- metadata records, the system suggests that the recommended records are relevant to the folder content
- communities, suggesting to the user to join communities, as they may deal with user related topics of interests
- users, suggesting to the user to enter in relationship with a user or give a glance at the publicly available documents of the recommended user, as he may have overlapping topics of interests.

Figure 2 User interface: recommendation folder



In the following we give an idea to the reader on how our recommendation algorithms work and devote special attention to the recommendation of users, which is indeed an important way in CYCLADES to relate users with similar interests to each other and help users to become aware of other users sharing similar interests.

The recommendation of collections to a folder consists in automatically determining the archives in which to search for relevant records to the folder. It is accomplished by means of a technique called automated source selection, see for e.g., Fuhr (1999). Roughly, it consists in

- The computation of an approximation of the content of each OAI compliant archive (statistical information about the content of an information source, see Callan and Connell, 2001), registered in CYCLADES.
- The selection of those archives deemed as most relevant to a folder, by relying on the approximations of the archives' content and on the folder profile. Both steps are done periodically, see Candela and Straccia (2003) for a detailed description.

The recommendation of records to a user folder consists in automatically determining the records deemed as relevant to a folder and is accomplished roughly in four steps as follows (see Avancini and Straccia, 2003 for a detailed description):

- select a set of most similar folders to the user folder, according to a similarity measure between folder profiles (see next section)
- from this set, determine a pool of possible recommendable records, which consists of the records belonging to the similar folders
- for each of the records in the pool compute a recommendation score
- select and recommend a subset of records with highest score and not yet recommended to the user folder.

The recommendation of communities to a user folder consists in automatically determining the communities, dealing about topics relevant to the topic of the user folder. It is accomplished roughly similar to the recommendation of records, the reader may see to Renda and Straccia (2002) for a more detailed description. While for the recommendation of records and collections, we have conducted experiments in order to evaluate the effectiveness of the adopted algorithms, this is not yet the case for the recommendation of communities, to which we will devote major effort in future work.

In the following section, we detail the algorithms used for user recommendation. We also report some experimental results of their effectiveness.

3 User recommendation and experimental evaluation

In the following, consider a set of users u_k, a set of folders F_i and a set of available metadata records d_j distributed in the folders. For ease, we consider a metadata record as a piece of plain text. Of course, more sophisticated algorithms can be devised by taking into account the metadata structure. Metadata records belong to folders and each user may also rate a document within a folder he has access to. With r_{ijk} we indicate the rating value given by a user u_k to record d_j, which is stored in folder F_i. We further assume that whenever a data item d_j belongs to a folder F_i of a user u_k, an implicit default rating r_{ijk} is assigned. Indeed, a record belonging to a folder of a user is an implicit indicator of being the record relevant to the user folder. Finally, we average the ratings r_{ijk} relative to the same folder-document pair (i,j) and indicate it as $rij = \text{mean}_{k>0}\ r_{ijk}$.

All records are indexed according to the well-known vector space model of Salton and McGill (1983). With $d_j = \langle w_{j1}, \ldots, w_{jm} \rangle$ we indicate its indexed representation, where $0 <= wjk < = 1$ is the 'weight' of term t_k in the record d_j. The folder profile, which is a machine representation of what a folder is about, denoted f_i, for folder F_i is computed as the centroid, or average, of the records belonging to F_i, i.e., $f_i = (1/|F_i|)\,\Sigma_{dj\varepsilon}\, F_i d_j$, thus, it is represented as a vector of weighted terms as well, i.e., $f_i = \langle w_{i1}, \ldots, w_{im} \rangle$. The user profile of a user u (denoted p_u) is built as the centroid of the user's folder profiles, i.e., if F_u is the set of folders belonging to the user u, then $pu = (1/|F_u|)\Sigma_{Fi\varepsilon Fu} f_i$,

Therefore, the user profile is represented as a vector of weighted terms as well, i.e., $p_u = \langle w_{u1}, \ldots, w_{um} \rangle$. By relying on the vector representation of records, folders profiles and user profiles, we can easily determine a similarity measure between them. Indeed, the similarity among two vectors (whether records, folder profiles or user profiles) is computed as content correlation (denoted CSim(.,.)) and is the well-known cosine angle among the two normalised (*norm-2*) vectors, i.e., it is the scalar product between two vectors, e.g., $\mathrm{CSim}(v_1, v_2) = \Sigma\, kw_{1k} w_{2k}$.

Another correlation among folders can be determined by taking the ratings issued by users into account only. This similarity is called rating similarity of two folders F_1 and F_2 (denoted RSim(F_1, F_2) and is determined using the Pearson correlation coefficient, see Breese et al. (1998), i.e., $\mathrm{RSIM}(F_1, F_2) = \sum_j (r_{1j} - \overline{r_1}) \bullet (r_{2j} - \overline{r_2}) \big/ \sigma_1 \bullet \sigma_2$, where $\overline{r_i}$ is the mean of the ratings $r_{i1} \ldots r_{im}$, and σ_i is their standard deviation.

The combined similarity between two folders is then obtained by taking into account the content similarity and the rating similarity. In what follows, the combined similarity or simply similarity (denoted Sim(F_1, F_2) between two folders F_1 and F_2 will be determined as a linear combination between their content similarity and their rating similarity, i.e., $\mathrm{Sim}(F_1, F_2) = \alpha \mathrm{CSim}(f_1, f_2) + (1 - \alpha)\ \mathrm{RSim}(F_1, F_2)$ where $0 <= \alpha <= 1$.

3.1 User recommendation algorithm

The goal of the user recommendation algorithm is, given a folder F_t (called target folder) of user u, to recommend to F_t (and, thus, to user u) those users, which by the system are thought to have overlapping interests with the topic addressed by the folder F_t (and, thus, may be related to user u). We have analysed four different algorithms, with increasing level of effectiveness. We start with the first one implemented in the CYCLADES system, the recommendation algorithm follows a four-step schema:

- Select the set $MS(F_t)$ of s-most similar folders to F_t, according to a similarity measures. We can use either CSim, RSim or the combination of both Sim, in Avancini and Straccia (2003). We have already observed that Sim($\alpha = 0.5$) has better effectiveness, so we use it here as well.
- From this set of similar folders, determine a pool $P_U(F_t)$ of candidate users to be recommended, i.e., let $P_U(F_t)$ be the set of users being owners of the folders in $MS(F_t)$.

Compute the *recommendation score* for each possible recommendable user, i.e., for each user $u_k \in P_U(F_t)$ determine the *user hits factor* (where $F_i \in u_k$ means that folder F_i is accessible by user u_k) $h(uk) = |\{F_i : F_i \varepsilon MS(F_t), F_i \varepsilon uk\}|$ i.e., the number of folders F_i judged as similar to the target folder F_t belonging to the same user u_k. For each user $u_k \in P_U(F_t)$ the recommendation score $s(F_t, u_k)$ is computed as follows: $s(F_t, u_k) = h(u_k) \cdot \sum_{F_i \in MS(F_t), F_i \in u_k} Sim(F_t, F_i)$.

- Recommend to folder F_t, the *top-n* ranked users, ranked according to the recommendation score.

  The intuition behind Step 3 is that the more a user appears among the owners of the *top-s* similar folders, the more he is considered as relevant to the target folder. The second algorithm is a variation of the first one in which Step 3 is replaced with:

- For each user $u_k \in P_U(F_t)$, consider the profile of u_k, p_{uk} and compute the recommendation score as the similarity between the user profile p_{uk} and the profile f of the target folder F_t, i.e., $s(F_t, uk) = \mathrm{CSim}(f, p_{uk})$.

The intuition here is to use the user profile of recommendable users $u_k \in P_U(F_t)$ directly in place of the folder profiles of similar folders.

The third algorithm does not consider the set of similar folders, but just compares the profile of the target folder against all user profiles, using CSim. Note that in this way, no ratings are taken into account. Therefore, we remove Step 1 and in Step 2, the pool of candidate users, $P_U(F_t)$, is given by all users known to the system and Step 3 is as in algorithm 2.

Finally, the fourth algorithm takes advantage of the hierarchical structure of the folders, while the previous methods do not care about this structure and consider the folders at the same level. In particular, it uses *Bayesian classifiers* (see, McCallum and Nigam, 1998) to build folder profiles and then uses the *Shrinkage method* to build user profiles (an introduction to shrinkage estimators is presented in Carlin and Louis, 1996).

This method works as follows. Let θ_{ik} be the naive Bayesian estimator for folder F_i and term t_k. To compute θ_{ik} estimator on each node of the hierarchy we use *maximum likelihood* estimation (ML), i.e.,

$$\bar{\theta}_{ik} = \sum_{d_j \in F_i} w_{kj} \Big/ \sum_{t_r \in \mathrm{T}} \sum_{d_j \in F_i} w_{rj} \tag{1}$$

where T is the set of all terms. Then we compute $P(F_i \mid \vec{d}_j)$, the probability of belonging of document d_j to folder F_i, through Bayes' theorem, i.e.,

$$P(F_i \mid \vec{d}_j) = \frac{P(\vec{d}_j \mid F_i) P(F_i)}{P(\vec{d}_j)} = \frac{P(\vec{d}_j \mid F_i) P(F_i)}{\sum_{F_r \in F} P(\vec{d}_j \mid F_r)} \tag{2}$$

where

$$P(\vec{d}_j \mid F_i) = P(n) \cdot \frac{n!}{w_{1j}! \cdot w_{2j}! \cdot \ldots \cdot w_{|T|j}!} \prod_{k=1}^{|T|} \theta_{ik}^{w_{kj}} \tag{3}$$

and $P(F_i)$ can be estimated as

$$\frac{1+|\{d_j \in F_i\}|}{|F|+\sum_{F_r \in F}|\{d_j \in F_r\}|}.$$

We compute the user profile estimators using the hierarchy information, i.e., we compute folder estimators in the path from each leaf user folder to the root (the home folder of the user), according to the shrinkage method. Then the user profile is a combination of the θ estimators of the folders of the user.

Formally, if F_i is a leaf user folder distant δ steps from the root, the shrinkage methods build more robust estimates of θ_{ik} (indicated as $\breve{\theta}_{ik}$) by interpolating the estimators θ_{ik} obtained for F_i with the ML estimates θ_{ik}^r, obtained for its ancestors folders $\pi^r(F_i)$, for $r \in \{1, \ldots, \delta\}$. This amounts to computing $\breve{\theta}_{ik} = \sum_{r=0}^{\delta+1} \lambda_i^r \theta_{ik}^r$ equation (4) where $\theta_{ik}^0 = \hat{\theta}_{ik}$, the $\hat{\theta}_{ik}^r$ estimates are obtained according to equation (1) and the λ_i^r is the interpolation weights, with $\sum_{r=0}^{\delta+1} \lambda_i^r = 1$. Note that equation (4) assumes the existence of a 'virtual' parent folder $\pi^{\delta+1}(F_i)$ of the root characterised by the uniform estimate, i.e., such that $\hat{\theta}_{ik}^{\delta+1} = 1/|T|$ for all $t_k \in T$; this is done in order to smooth the parameters for those terms that are rare also in the user root (home) folder (i.e., in the entire training set), and eliminates the need for Laplace smoothing. The λ_i^r weights are determined by applying a variant of the expectation maximisation (EM) algorithm on a validation set.

3.2 Experimental evaluation

We tested our recommendation algorithm for effectiveness. Indeed, we first determine a corpus of data. From the corpus we select a test set of pairs (F_i, u_k), where F_i is the target folder and u_k is a user having folder F_i in his folder hierarchy. Second, user recommendations are given for each of the folders of the test set. Finally, we analyse the results.

As to date, neither is there a significant corpus within the CYCLADES system built by real users (to date, CYCLADES has 59 users sharing 118 folders, which contain 284 records with 35 non-default ratings), nor was it available during the development phase to 'tune' our algorithms, nor there exists an available corpus from the literature, which fits to our setting, with which we can build a corpus automatically.

Corpus

The corpus was selected from the ODP or DMOZ (http://dmoz.org). The ODP is the largest human-edited directory of the web. The ODP data includes over 3.8 million sites, about 60,000 editors and over 460,000 categories. The ODP powers the core directory services for the web's largest search engines and portals, e.g., Google (http://www.google.com). Each category in ODP contains a set of web documents, which have been evaluated by one or more editors for their relevance to the category. Furthermore, to each document within a category, Google assigns a score, using the PageRank of Brin and Page (1998) algorithm. We construct our corpus as follows. The set of users is the set editors of ODP. The set of records is the set of documents in

ODP. The set of folders is the set of categories in ODP. To each record d_j in folder F_i, evaluated by user u_k, we set the rating r_{ijk} equal to the PageRank score s_{ij} assigned to record d_j w.r.t. folder F_i. This means that r_{ij}, the average rating over all users rating records d_j in folder F_i, is indeed s_{ij}, note that all users rate $d_j \, \varepsilon \, F_i$ equally. But this does not matter us, as in the recommendation algorithm just the mean r_{ij} is required. To limit the amount of data, we considered all the categories under 'Science/Math' only, together with the involved records and users. The profiles of the folders have been restricted to the top weighted 100 terms. Totally, we have 500 folders distributed among 46 users, sharing 8083 documents, each having at least one rating associated. As the initial set of folders a user has access is small, we constructed a random algorithm, which takes a folder F and a user u having access to F, randomly selects a subtree of F to which u has access as well. For *Sim*, we used $\alpha = 0.5$ and $s = 100$.

Test set

To create the test set, we considered the set of all users U of the corpus, which have access to at least two folders. For each of these users $u_k \, \varepsilon \, U$ (46 users), we randomly choose a folder F_i of u_k. The set of chosen folders $F_{te} = \{F_i\}$ (35 folders) and the users u_k, i.e., (F_i, u_k) forms the test set. Totally, we have 46 users to be recommended distributed over 35 folders.

Evaluation

For each target folder $F_i \, \varepsilon \, F_{ts}$, where ($F_i$, u_k) belongs to the test set, we compute the set of recommended users u_k and ranked them according to their recommendation score p_{uk}. In this rank, we highlight the rank position of user u_k. If the recommendation score is 0, user u_k is ranked 0. Additionally, we determine their average rank, precision and recall. Totally, we performed 35 tests (number of folders).

Experimental results show that algorithm 4 outperforms all other algorithms, and that algorithm 3 performs at the second place, algorithm 2 is third, while the less effective one is algorithm 1. For all four algorithms, see Table 1, we computed recall and precision, by recommending the *top-n* users for each target folder, where $n \, \varepsilon \, \{1,2,5,10\}$. Precision is the fraction between correctly recommended users and the total amount of recommended users, i.e., $Precision_n = (|CorRec_n|)/(35n)$, where $CorRec_n$ is the set of correctly recommended users among all 35 tests and n is the number of recommended users for each test. Recall, is the fraction between correctly recommended users and the number of test users, i.e., $Recall_n = (|CorRec_n|)/46$.

Of course, the more users we recommend (i.e. n increases), the more correctly recommended users we have (recall improves), but the less precise we are (precision decreases). We also report the standard measure $F_{1n} = (2 \, Precision_n \, Recall_n)/(Precision_n + Recall_n)$, which gives us an estimate of the combination between precision and recall. It turns out that algorithm 4 is the most effective one and, interestingly, that recommending just the top ranked user is the most satisfactory compromise between precision and recall. This result is as we expected as the fourth algorithm takes heavily into account the hierarchical structure of the folders.

Table 1 Effectiveness measures

	Algorithm	*Top-1*	*Top-2*	*Top-5*	*Top-10*
Precision	1	0.03	0.03	0.03	0.03
	2	*0.23*	*0.16*	*0.08*	*0.05*
	3	0.40	0.24	0.11	0.07
	4	*0.54*	*0.36*	*0.21*	*0.12*
Recall	1	0.02	0.04	0.11	0.24
	2	*0.17*	*0.24*	*0.30*	*0.39*
	3	0.30	0.37	0.43	0.52
	4	*0.37*	*0.43*	*0.58*	*0.71*
F1	1	0.02	0.03	0.05	0.06
	2	*0.20*	*0.19*	*0.13*	*0.09*
	3	0.35	0.29	0.18	0.12
	4	*0.44*	*0.39*	*0.31*	*0.20*

Average rank for algorithm: 1 (10.94); 2 (3.11), 3 (2.58), 4 (2.01).

4 Conclusions

In this paper, we describe the Digital Library environment CYCLADES, which is not only an information source where users may submit queries to searching for relevant information, but also a personalised, and more importantly, a collaborative working and meeting space. The functionality provided by the CYCLADES system, can be organised into four categories: users may

- Search for information, not only by means of generic queries, but also by taking into account the learned user topics of interests.
- Organise the information space according to the folder and personalised collections paradigm, which allow users to personalise the information space made available within CYCLADES.
- Collaborate, in shared working space, with other users, which may have similar interests or more generally are related according to some purpose.
- Get recommendations from the CYCLADES system. CYCLADES not only provides recommendation of records, as it usually happens in personalisation system dealing with documents, but by taking advantage of the highly collaborative environment, it may also recommend communities, collections and users as well. Particular attention has been paid to the user recommendation part and some experiments showing its effectiveness.

Acknowledgment

This work was funded by the European Community in the context of the CYCLADES project IST-2000-25456, under the Information Societies Technology Programme.

References

Avancini, H. and Straccia, U. (2003) *Personalization, Collaboration, and Recommendation in the Digital Library Environment CYCLADES*, Technical Report ISTI-CNR. 2003-TR-59.

Bentley, R. *et al*. (1997) 'Basic support for cooperative work on the world wide web', *International Journal of Human Computer Studies*, Vol. 46, pp.827–846.

Bollacker, K. *et al*. (1999) 'A system for automatic personalized tracking of scientific literature on the web', *Proceedings of Digital Libraries 99 – The Fourth ACM Conference on Digital Libraries*, ACM Press, New York, pp.105–113.

Breese, J. *et al*. (1998) 'Empirical analysis of predictive algorithms for collaborative filtering', *Proceedings of the 14th Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, Madison, Wisconsin, USA, pp.43–52.

Brin, S. and Page, L. (1998) 'The anatomy of a large-scale hypertextual Web search engine', *Computer Networks and ISDN Systems*, Vol. 30, Nos. 1–7, pp.107–117.

Callan, J. and Connell, M. (2001) 'Query-based sampling of text databases', *ACM Transactions on Information Systems*, Vol. 19, No. 2, pp.97–130.

Candela, L. and Straccia, U. (2003) 'The personalized, collaborative digital library environment CYCLADES and its collections management', in Retrieval, J., Callan, F., Crestani and Sanderson, M. (Eds.): *Proceedings of Multimedia Distributed Information*, Springer Verlag.

Carlin, B. and Louis, T. (1996) *Bayes and Empirical Bayes Methods for Data Analysis*, Chapman and Hall.

Fernandez, L. *et al*. (2000) 'Mibiblio: personal spaces in a digital library universe', *ACM Digital Libraries*, pp.232–233.

Fox, E. and Marchionini, G. (2001) 'Digital libraries: introduction', *Communications of the ACM*, Vol. 44, No. 5, pp.30–32.

Fuhr, N. (1999) 'A decision-theoretic approach to database selection in networked IR', *ACM Transactions on Information Systems*, Vol. 3, No. 17, pp.229–249.

McCallum, A. and Nigam, K. (1998) 'A comparison of event models for naive Bayes text classification', *Proceedings of the 1st AAAI Workshop on Learning for Text Categorization*, Madison, US.

Renda, M. and Straccia, U. (2002) 'A personalized collaborative digital library environment', *Proceedings of 5th International Conference on Asian Digital Libraries (ICADL-02), n.2555 in Lecture Notes in Computer Science*, Springer-Verlag, Singapore, Republic of Singapore, pp.262–274.

Salton, G. and McGill, J. (1983) *Introduction to Modern Information Retrieval*, Addison Wesley Publ. Co., Reading, Massachussetts.