A Multilevel Approach to Intelligent Information Filtering: Model, System, and Evaluation

J. MOSTAFA
Indiana University
S. MUKHOPADHYAY
Purdue University
W. LAM
The Chinese University of Hong Kong
and
M. PALAKAL
Purdue University

In information-filtering environments, uncertainties associated with changing interests of the user and the dynamic document stream must be handled efficiently. In this article, a filtering model is proposed that decomposes the overall task into subsystem functionalities and highlights the need for multiple adaptation techniques to cope with uncertainties. A filtering system, SIFTER, has been implemented based on the model, using established techniques in information retrieval and artificial intelligence. These techniques include document representation by a vector-space model, document classification by unsupervised learning, and user modeling by reinforcement learning. The system can filter information based on content and a user's specific interests. The user's interests are automatically learned with only limited user intervention in the form of optional relevance feedback for documents. We also describe experimental studies conducted with SIFTER to filter computer and information science documents collected from the Internet and commercial database services. The experimental results demonstrate that the system performs very well in filtering documents in a realistic problem setting.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*clustering*; *selection process*; I.2.6 [**Artificial Intelligence**]: Learning; I.7.3 [**Text Processing**]: Index Generation

S. Mukhopadhyay was partially supported by NSF CAREER grant ECS-9623971 during the course of the research reported in this article.
Authors' addresses: J. Mostafa, School of Library and Information Science, Indiana University, Bloomington, IN 47405-1801; email: jm@juliet.ucs.indiana.edu; S. Mukhopadhyay and M. Palakal, Computer and Information Science, Purdue University School of Science at Indianapolis, Indianapolis, IN 46202; W. Lam, Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, Hong Kong.

ACM Transactions on Information Systems, Vol. 15, No. 4, October 1997, Pages 368–399.

General Terms: Algorithms, Experimentation, Theory

Additional Key Words and Phrases: Automated document representation, information filtering, user modeling

1. INTRODUCTION

Information-filtering (IF) systems have recently gained popularity, mainly as part of various information services based on the Internet [Edwards et al. 1996; Oard 1996]. These systems are similar to conventional information retrieval (IR) systems in that they aid in selecting documents that satisfy users' information needs. However, certain fundamental differences do exist between IF and IR systems, making IF systems interesting and an independent object of analysis [Belkin and Croft 1992]. IR systems are usually designed to facilitate rapid retrieval of information units for relatively short-term needs of a diverse population of users. In contrast, IF systems are commonly personalized to support long-term information needs of a particular user or a group of users with similar needs. They accomplish the goal of personalization by directly or indirectly acquiring information from the user. In IF systems, these long-term information needs are represented as interest profiles (Lewis [1992a] refers to them as standing queries), which are subsequently used for matching or ranking purposes. The interest profiles are maintained beyond a single session and may be modified based on users' feedback. Another important difference has to do with the document source. IR systems usually operate on a relatively static set of documents, whereas IF systems are usually concerned with identifying relevant documents from a continuously changing document stream.

To operate efficiently, IF systems must acquire and maintain accurate knowledge regarding documents as well as users. The dynamic nature of users' interests and the document stream makes the maintenance of such knowledge quite complex. Acquiring correct user interest profiles is difficult, since users may be unsure of their interests and may not wish to invest a great deal of effort in creating such a profile. Acquiring information regarding documents is also difficult, because of the size of the document stream and the computational demands associated with parsing voluminous texts. At any time, new topics may be introduced in the document stream, or user's interests related to topics may change. Furthermore, sufficiently representative documents may not be available to facilitate a priori analysis or training. Research on filtering, so far, has not clarified to a significant extent how these particular problems associated with users and documents may influence the overall filtering process.

In this article, we present both an analytical and an empirical examination of the basic problems in filtering. In our investigation here of the demands placed on IF systems, we identify the relevant functions and express them at a suitable abstraction level. This abstraction (we refer to it as the *model*) is then implemented as a system using well-known tech-

niques from information science and machine learning. Following this, the performance of the resulting system is subjected to rigorous experimental analysis to clarify the influence of major constituent functions on the overall filtering process. The primary objective of an IF system is to perform a mapping from a space of documents to a space of user relevance values. This mapping, in turn, can be decomposed into a multilevel process, where the intermediate functions involve the subproblems of representation, classification, and profile management. To ensure effective service, we further assume that these functions must be realized under two strict constraints. First, user intervention in the operation of the system must be minimized. That is, the system should rely on automated techniques as much as possible for acquiring information about documents and users. Second, when faced with changes in documents or users' information needs, the system must adjust quickly with little or no degradation in performance.

In the rest of this section, we discuss in more detail the challenges associated with performing effective filtering while minimizing user intervention and system degradation. We then identify some of the basic problems associated with filtering and delineate our approach for addressing them. We conclude the section by surveying related research. In Section 2, we present our model for information filtering. A description of an implementation of the model, named SIFTER (Smart Information Filtering Technology for Electronic Resources), is provided in Section 3. Results of experimental analysis conducted on SIFTER (and indirectly on the underlying model used) are presented in Section 4. In Section 5, we discuss possible future extensions of SIFTER. Finally, we present our conclusions in Section 6.

1.1 Problem Description

Uncertainties in the filtering environment—especially the dynamic nature of users' interests and the document stream—make it extremely difficult to gather and maintain accurate information necessary for filtering. Rapid or gradual changes introduced in the environment, viewed from the perspective of the filtering system, are sources of uncertainty. To manage such uncertainties requires a high level of adaptivity on the system's part. This adaptivity can be achieved by applying various machine-learning techniques. The overall problem of IF may then be broadly posed as learning a map from a space of documents to the space of real-valued user relevance factors. More precisely, denoting the space of documents as $\mathcal{D}$, the objective is to learn a map $f : \mathcal{D} \rightarrow \mathbb{R}$ such that $f(d)$ corresponds to the relevance of a document d. Given that such a map is known for all points in $\mathcal{D}$, a finite set of documents can always be rank-ordered and presented in a prioritized fashion to the user.

In an IF system, f is not known a priori and has to be estimated on-line based on queries and user feedback. This could, in principle, be accomplished by setting up some form of a parameterized map approximator

(such as artificial neural networks) and updating the parameters based on the feedback. Such a direct on-line learning of the map f, however, is computationally intensive and requires a large number of user feedbacks, considering the high dimensionality of any reasonable representation of the documents. To provide a practically feasible solution to the filtering problem, we decompose the latter into two levels. The higher level represents a classification mapping f_1 from the document space to a finite number of classes $\{C_1, \ldots, C_m\}$ (i.e., $f_1 : \mathcal{D} \rightarrow \{C_1, \ldots, C_m\}$). This mapping is learned in an off-line setting, based on a representative database of documents, either by using *prior* information concerning the classes and examples or by automatically discovering abstractions using a clustering technique. Hence, this higher level partitions the document space into m equivalent classes over which user relevance is estimated. The lower level subsequently estimates the mapping f_2 describing user relevance for the different classes (i.e., $f_2 : \{C_1, \ldots, C_m\} \rightarrow \mathbb{R}$). Since f_2, unlike f and f_1, deals with a finite input set of relatively few classes, the on-line learning of f_2 is not unrealistically time consuming and burdensome on the user. Thus, the map f is being learned as the composition of f_1 and f_2. The decomposition of f into f_1 and f_2 clearly limits the maximum achievable filtering accuracy, since a class may not correspond to a constant user interest. However, in our experience, the resulting inaccuracy is more than adequately compensated for by the substantial reduction in learning complexity. If greater accuracy is desired, it can be achieved as a two-stage process. In the first stage, a two-level map (i.e., f_1 and f_2) is learned as stated before. Subsequently, a more general single-level learning scheme can be initialized on the basis of learned f_1 and f_2. From then onward, the general map can be used for ranking purposes and can be updated on the basis of user feedback.

Decomposition of f only aids in reducing the learning complexity; it does not eliminate it. The on-line learning problem is made even more difficult due to the following factors:

(1) *Difficulty of Representation*: In general, it is not possible to represent $\mathcal{D}$ exactly by a finite-dimensional space that corresponds to some features of the documents (e.g., the relative frequencies of some predefined keywords). Hence, any finite-dimensional representation space $\mathcal{D}'$ is merely an approximation to $\mathcal{D}$, and there is always a loss of information in the process. The area of document representation and indexing [Salton and McGill 1983] is devoted to discovering methods for finite-dimensional representations that minimize the information loss in some sense. In a dynamic environment, to make the problem more difficult, the most preferable representation scheme is also a function of time. The choice of the representation scheme directly affects the realization of function f_1.

(2) *Stochasticity of Feedback*: The user relevance feedback may at certain times appear to be random to the filtering system. This can occur due to several reasons. First, the particular user interacting with the system

may have uncertain needs or may not be very discriminating in expressing his or her needs. Second, depending on the f_1 chosen, the target classes may not correspond to the way a user would normally group documents. This may lead to the generation of different user relevance feedback values for documents belonging to the same class. The third and final factor relates to the difficulty described in (1). On certain occasions, user feedback may be motivated by particular features (e.g., keywords) in documents that are actually not part of the underlying representation scheme. Feedback generated based on such "missing features" would appear as random, because the system would be unable to determine what caused such feedback.

(3) *Changing Interests of the User*: Due to personal or professional reasons, a user's interests may shift or change. These changes may happen in a relatively short duration of time or over a long period. We refer to all such situations as the nonstationary user case. The shifts can affect the user's interests partially or fully. Whatever the scope of such shifts, the interest profile must be updated accordingly. The map f_2 is directly affected by this problem.

As mentioned earlier, due to the inherent complexity, filtering based on a direct learning approach is very difficult to accomplish in an efficient fashion. Decomposition allows us to isolate more specific problems, and we solve them by relying on existing and newly developed approaches. The main contributions of this article can now be summarized as follows:

—We present a general model of filtering. As a way to reduce complexity, the architecture of the model incorporates multilevel functional decomposition and supports generality through modularity. It admits application of virtually any preferred techniques for basic tasks involving representation, classification, and profile management.

—The idea of learning is made central to the filtering process. We show how learning techniques can support the high degree of adaptivity required while minimizing user intervention. We apply learning techniques for acquiring information about both documents and users. To support adaptation to changes in the document stream, an unsupervised cluster discovery method is used. A reinforcement learning algorithm with very low overhead is used for user interest profile acquisition.

—We demonstrate how representation can be conducted on a dynamic stream of text. The method provides a high degree of control in determining what content to capture and what to ignore. The classification process is also designed to be flexible. The set of classes (i.e., the target of f_1) can easily be changed by invocation of a relearning process. Both of these features allow convenient tuning of the filter to minimize user intervention.

—We describe a method to handle profile degradation due to shifts in user interests. Graceful handling of interest shifts without requiring additional data from the user is supported by the method. It is capable of

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.