UNIFIED PATENTS

EXHIBIT 1012

In *Proceedings of the 1995 USENIX Technical Conference*, pages 177-86, 1995

SIFT – A Tool for Wide-Area Information Dissemination [*]

Tak W. Yan Hector Garcia-Molina
Department of Computer Science
Stanford University
Stanford, CA 94305
{tyan, hector}@cs.stanford.edu

February 16, 1995

Abstract

The dissemination model is becoming increasingly important in wide-area information system. In this model, the user subscribes to an information dissemination service by submitting profiles that describe his interests. He then passively receives new, filtered information. The Stanford Information Filtering Tool (SIFT) is a tool to help provide such service. It supports full-text filtering using well-known information retrieval models. The SIFT filtering engine implements novel indexing techniques, capable of processing large volumes of information against a large number of profiles. It runs on several major Unix platforms and is freely available to the public. In this paper we present SIFT's approach to user interest modeling and user-server communication. We demonstrate the processing capability of SIFT by describing a running server that disseminates USENET News. We present an empirical study of SIFT's performance, examining its main memory requirement and ability to scale with information volume and user population.

1 Introduction

Technological advances have made wide-area information sharing commonplace. A suite of tools have emerged for network information finding and discovery; e.g., Wide-Area Information Servers (WAIS) [KM91], archie [ED92], World-Wide Web (WWW) [BLCGP92], and gopher [McC92]. However, these new tools have one important missing element. They provide a means to search for existing information, but lack a mechanism for continuously informing the user of new information. The exploding volume of digital information makes it difficult for the user, equipped with only search capability, to keep up with the fast pace of information generation. Instead of making the user go after the information, it is desirable to have information selectively flow to the user. In an *information dissemination* (aka. *alert, information filtering, selective dissemination of information*) service, the user expresses his interests in a number of long-term, continuously evaluated queries, called *profiles*. He will then passively receive documents filtered according to the profiles. Such a service will become increasingly important and form an indispensable tool for the dynamic environment of wide-area information systems.

A very simple kind of information dissemination service is already available on the Internet: mailing lists (see e.g., [Kro92]). Hundreds of mailing lists exist, covering a wide variety of topics. The user subscribes to lists of interest to him and receives messages on the topic via email. He may also send messages to the lists to reach other subscribers. LISTSERV is a software system for maintaining mailing lists. A problem with the mailing list mechanism as a tool for information dissemination is that it provides a crude granularity of interest matching. A user whose information need does not exactly match certain lists will either receive too many irrelevant or too few relevant messages. The USENET News (or Net-

[*] This research was sponsored by the Advanced Research Projects Agency (ARPA) of the Department of Defense under Grant No.MDA972-92-J-1029 with the Corporation for National Research Initiatives (CNRI). The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsement, either expressed or implied, of ARPA, the U.S. Government, or CNRI.

news) system [Kro92], an electronic bulletin board system on the Internet, is similar in nature to mailing lists. While Netnews is extremely successful with millions of users and megabytes of daily traffic, at the same time it often creates an information overload. Like mailing lists, the coarse classification of topics into newsgroups means that a user subscribing to certain newsgroups may not find all articles interesting, and also he will miss relevant articles posted in newsgroups that he does not subscribe to.

Recent research efforts on information filtering focus on the filtering *effectiveness*, attempting to provide more fine-grained filtering using relational, rule-based, information retrieval (IR), and artificial intelligence approaches. With few exceptions, they are often small scale (i.e., involving a small number of users or profiles) and thus the need to provide *efficient* filtering is not apparent. However, in a large-scale wide-area system where the number of information providers and seekers are large, efficiency in the dissemination process is an important issue and must be addressed.

The Stanford Information Filtering Tool (SIFT) is a tool for information providers to perform large-scale information dissemination. It can be used to set up a clearinghouse service that gathers large amount of information and selectively disseminates the information to a large population of users. It supports full-text filtering, using well-known and well-studied IR models. The SIFT filtering engine implements novel indexing techniques, capable of scaling to large number of documents and profiles. It runs on several major Unix platforms and is freely available to the public by anonymous ftp at URL:

```
ftp://db.stanford.edu/pub/sift/sift-1.0.tar.Z
```

In this paper we describe SIFT. We present its approach to user interest modeling and user-server communication. We demonstrate the processing capability of SIFT by describing a running server that disseminates tens of thousands of Netnews articles daily to some 13,000 subscriptions. We describe the implementation of SIFT, focusing on the filtering engine. Finally we present an empirical study of SIFT's performance, examining its main memory requirement and ability to scale with information volume and user population.

2 Other Previous Work

Boston Community Information System [GBBL85] is an experimental information dissemination system. Like SIFT, it allows a finer granularity of interest matching than mailing lists or Netnews. Users can express their interests with IR-style, keyword-based profiles. The system broadcasts new information via radio channel to all users, who then apply their own filters locally. While the radio communication channel makes broadcast inexpensive, local processing of mostly irrelevant information is very expensive. (For every user, a personal computer is dedicated for this purpose – this is cited as a major source of complaints from the users.)

Information Lens [MGT+87] provides categorization and filtering of semi-structured messages such as email. The user defines rules for filtering, and the processing is done at the user site. It provides effective filtering, but local processing is expensive for large-scale information dissemination. Similarly, the "kill file" mechanism in certain news reader programs allows the user to locally screen out irrelevant articles. A kill file only removes specified articles from newsgroups that a user subscribes to, but it does not discover relevant articles in other newsgroups. To provide the same kind of filtering power as SIFT would require much local processing. It is more cost-effective to pool profiles together to share the processing overhead.

The Tapestry system [GNOT92] is a research prototype that uses the relational model for matching user interests and documents; filtering computation is done not on the properties of individual documents, but rather on the entire append-only document database. Efficient query processing techniques are proposed for handling this kind of queries. Tapestry is built on top of a commercial relational database system. The Pasadena system [WF91] investigates the effectiveness of different IR techniques in filtering. It collects new documents from several Internet information sources, and periodically run profiles against them. Similar to SIFT, it uses a clearinghouse approach, but efficient filtering is not addressed.

In [YGM94b, YGM94c] we proposed a variety of indexing techniques for speeding up information filtering under IR models. There we evaluated these techniques using analysis and simulation. The SIFT filtering engine is a real implementation of one class of index structures that we found to be efficient.

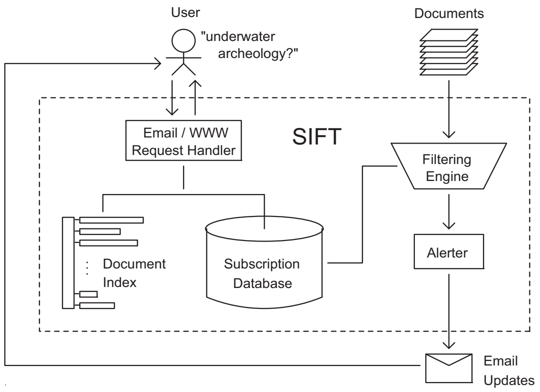

Figure 1: An overview of SIFT

3 SIFT

We begin our presentation of SIFT with an example. Suppose a user is interested in underwater archeology (Figure 1). He sends an email subscribe request to a SIFT server specifying the profile "underwater archeology," and optionally parameters that control, for example, how often he wants to be updated or how long his subscription is for. He may alternatively access the SIFT server via WWW, using a graphical WWW client interface to fill out a form with the subscription information. (Before he subscribes, he may test run his profile against an index of a sample document collection.) His subscription is stored in the subscription database. As the SIFT server receives new documents, the filtering engine will process them against the stored subscriptions, and notifications will be sent out based on the user-specified parameters.

In the following we detail the SIFT system from the perspectives of user interest modeling and communication protocol. We then illustrate SIFT using the Netnews SIFT server.

3.1 User Interest Modeling

A user subscribes to a SIFT server with one or more subscriptions, one for each topic of interest. A subscription includes an IR-style profile, and as mentioned additional parameters to control the frequency of updates, the amount of information to receive, and the length of the subscription. A subscription is identified by the email address of the user and a subscription identifier.

3.1.1 Filtering Model

The interest profile can be expressed in one of two IR models: *boolean* and *vector space* [Sal89]. We first focus on the vector space model.

In vector space model, queries and documents are identified by terms, usually words. If there are m terms for content identification, then a document D is conceptually represented as an m-dimensional vector $D = \langle w_1, w_2, \ldots, w_m \rangle$, where weight w_i for term t_i signifies its statistical importance, such as its frequency in the document. We may also write D as $\langle (t_{i_1}, w_{i_1}), \ldots, (t_{i_k}, w_{i_k}) \rangle$ where $w_{i_j} \neq 0$; e.g., $\langle$ (underwater, 60), (archeology, 60) $\rangle$. A query is similarly represented. For a document-query pair, a similarity measure (such as the dot product) can be computed to determine how "similar" the two are. In an IR environment, the top ranked documents are retrieved for a query.

In an information filtering setting, a key question is how many documents to return to the user. We could have allowed the user to receive a fixed number of top ranked documents per update period, e.g., as done in [FD92]. However, this is not very desirable: in a period when there are many relevant documents, he may miss some (low *recall*[1]); in a period when there are few interesting documents, he may receive irrelevant ones (low *precision*[1]). We instead allow the user to specify a *relevance threshold*, which is the minimum similarity score that a document must have against the profile for it to be delivered. A default value is supplied to the user for convenience.

Instead of using the vector space model, the user may use boolean profiles to specify words that he wants in documents received, and words to be excluded. For example, the boolean profile "fly fishing not underwater" is for documents that contain both words "fly" and "fishing" but not the word "underwater." The reader may note that the SIFT boolean model only allows conjunction and negation of words. However, the user may approximate disjunction semantics by submitting multiple subscriptions (though a document may match more than one subscriptions).

3.1.2 Profile Construction and Modification

To assist the user with the construction of a profile, a SIFT server provides a test run facility. The user may

[1] Recall is the proportion of relevant documents returned, and precision is the proportion of documents returned that are relevant.

run his initial profile against an existing, representative collection of documents to test the effectiveness of his profile. He may interactively change the profile and (for weighted profiles) adjust the threshold to the desired level of precision and recall. When he is satisfied with the performance of the filtering, he may then subscribe with the selected settings.

After the user receives some periodic updates, he may decide to modify his profile or change the threshold for vector space profiles. He may do this by accessing the SIFT server. Furthermore, for vector space profiles, *relevance feedback* [Sal89], a well-known technique in IR to improve retrieval effectiveness, can be used. The user simply gives SIFT the documents that he finds interesting; after examining them, the server adjusts the weights of the words in the user's profile accordingly.

3.2 Communication Protocol

There are two modes of communication between the user and a SIFT server. In the interactive mode, the user subscribes, test-runs a profile, views, updates, or cancels his subscriptions. In the passive mode, the user periodically receives information updates. Instead of developing a new communication protocol, we make use of current technologies: email [Cro82] and World-Wide Web HTTP [BLCGP92].

First we discuss the interactive communication mode. Email communication is the lowest common denominator of network connectivity. By having an email interface, a SIFT server is accessible from users with less powerful machines, with limited network capability, or behind Internet-access firewalls. We adopt the LISTSERV mailing list syntax wherever possible, to ensure that minimal learning is required. We also use default settings to reduce the complexity of email requests for novice users.

As the appeal of hypermedia navigation and the development of sophisticated client interfaces are making WWW the preferred tool for wide-area information sharing, we also developed a WWW access interface for SIFT. Using a WWW client program, the user interacts with the SIFT server through a user-friendly graphical interface. We believe the dual email and WWW access covers the tradeoff between wide availability and ease of use.

In the passive mode of user notification, a SIFT server sends out email messages that contain excerpts of new, potentially relevant documents (certain number of lines from the beginning, as specified by the user). After the user reads the excerpts, he may access the SIFT server to retrieve the entire documents. Currently the excerpts are formatted to be read from regular mail readers. We plan to offer the option of formatting them in HTML, so that the user may view the notifications from sophisticated WWW viewers, and then interactively retrieve interesting documents from SIFT or provide feedback via HTTP.

3.3 SIFTing Netnews

Using SIFT, we have set up a server for selectively disseminating Netnews articles (text articles only; binary ones are first screened out). Like any other SIFT server, the user accesses the Netnews SIFT server via email or WWW. The reader is encouraged to try it out: for email access, please send an electronic message to `netnews@db.stanford.edu`, with the word "help" in the body; for WWW access, please connect to `http://sift.stanford.edu`. In February 1994, we publicized the Netnews server in two newsgroups; within ten days of the announcement, we received well over a thousand profiles. The number of profiles keeps increasing and now (November 1994) exceeds 13,000, submitted by users from almost all continents. Table 1 shows some interesting statistics obtained from the server on the day of November 10, 1994. The average number of articles is over the week of November 4 - 10, 1994.

Number of subscriptions	13,381
Number of users	5,146
Daily average number of articles	45,127
Average notification period (in days)	1.4

Table 1: Netnews SIFT server statistics

Apparent from these numbers is that the load on the SIFT server is very high. It is necessary to match an average of over 45,000 articles against some 13,000 profiles and deliver most updates within a day. The efficient implementation of the SIFT filtering engine enables the job to be done on regular hardware, a DECstation 5000/240. Even though we believe SIFT is efficient, there is a limit to the load that it can handle. It is necessary to replicate the server; in fact, we have been in contact with several sites (in the U.S. and Europe) that expressed interests in providing the same service.

The Netnews SIFT server is not meant to be a replacement of the Netnews system, which is an in-

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.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**