MIT9904-08
Haystack: Per-User Information Environments

David Karger and Lynn Andrea Stein

I. Project Overview

The Haystack project is investigating the ways in which electronic infrastructure can be used to triangulate among the different knowledges of an individual, of collegial communities to which we belong, and of the world at large. Its infrastructure consists of a community of independent, interacting information repositories, each customized to its individual user. It provides automated data gathering (through active observation of user activity), customized information collection, and adaptation to individual query needs. It also facilities inter-haystack collaboration, exposing (public subsets of) the tremendous wealth of individual knowledge that each of us currently has locked up in our personal information spaces. The Haystack-NTT project involves augmenting its customization, learning and adaptation, and inter-haystack communication.

Work to date has focused on building infrastructure: putting together the basic tool that collects a user's information, stores it in a sensible data model, and offers basic search and adaption capabilities. With this infrastructure mostly stable, we can begin to explore the opportunities it creates for system adaptation and for collaborative interaction—both among haystacks and between haystacks and other information retrieval systems.

II. Research Plan for 2000-2001

We propose to extend current work on Haystack in the following directions:

1 - Beginning to triangulate between the knowledge of individual users, collegial communities, and the world knowledge represented by the World Wide Web.

Until now, work on Haystack has focused on building and adapting individuals' personal corpora. We will now begin to explore the ways that information in these corpora can be leveraged to deal with other corpora. We will use the individual corpora represented by single Haystacks as filters on queries to traditional Web engines. This involves work on both pre-processing (query expansion, based on the user's likely intent as represented by his/her Haystack) and post-processing (query refinement, based on the user's likely preferences, again as represented by the Haystack). This project involves building both query expansion and query refinement modules as well as empirical work on the relative benefits of these modules in actual user searches.

2 - Development and integration of a uniform model for structured (relational-database style) and unstructured (English text) queries.

We are in the process of integrating a database system into Haystack, which will allow us to more meaningfully query the non-textual, relational information (such as dates, authors, and document similarity measures) that our Haystack system currently captures. We will explore ways to integrate such relational queries with traditional text-search. Traditional databases are very black and white: an object either matches or does not match a given query. Using Haystack's adaptive techniques, we hope to introduce "fuzzier" (and thus more forgiving of user inaccuracies) relational searching. For example, on a database query we might also return objects "similar" to the ones that actually match the query. More generally, matches to particular database queries can be treated as "features," much the way the presence or absence of a given word is treated as a feature, and the traditional tool of information retrieval and machine learning, such as relevance feedback and clustering, can be applied to deal with such features. This work builds directly on the structured query and database back end tasks completed in this project's first year.

3 - Clustering as a means of organizing user data and improving query result presentation.

Professor Karger has been involved in previous work at Xerox Parc exploring the use of clustering as a tool to let users navigate large sets of documents. We aim to explore this technology within Haystack and exploit the adaptive nature of Haystack to increase the power of clustering. For example, while traditional document browsing systems construct their clusters automatically, Haystack provides an environment in which the user's response to the clustering presented can be used to improve the cluster structure. For example, the user can move objects that were not placed correctly by the clustering system, and the clusterer can use this "training data" to shift other objects as well. As another example, the system can notice which clusters the user actually examines, and use this observation to "rebalance" the cluster sizes so that all clusters are presented at an appropriate level of detail.

Ultimately, we feel that a cluster-based system can supplant the directory hierarchy as the standard form of information structuring in the users file system.

4 - Building a better GUI.

The extra features we are building into our system require a more sophisticated user interface. We will build an "object oriented user interface" that provides a convenient way to modify the display for all the distinct object types being stored in user Haystacks. We will provide tools that let users effectively edit, rather than just navigate, the data model. We will develop techniques to present clustered object sets to the user and collect user modifications of those clusterings for use in adapting the clustering. Such a cluster-based user interface may replace the fixed files-and-folders interface currently in use.

5 - Inter-haystack communication.

We have begun discussion with the W3C on ways to let Haystacks import data from and export data to other RDF-aware systems. The immediate application is to let two haystacks communicate, but we will develop with the intent to interface with other systems using a more general protocol. This task involves constructing a haystack interchange language; some of this work will be undertaken in connection with the Semantic Web project currently being pursued by the W3C.