Haystack: Per-User Information Environments
MIT9904-08

Progress Report: July 1, 2001—December 31, 2001

David Karger

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.

The Haystack project stands upon three research pillars. We study semi-structured databases and semantic-web ontologies as tools for representing all of the knowledge useful to an individual. We explore user interfaces that present this rich information to the user in an effective way, letting them search, navigate, and manipulate their information. And we investigate deductive agents and machine learning as tools to reduce an individuals information management burden.

Progress Through December 2001

As discussed in the previous report, the first half of the year was focused on developing our “RDF triple store” that is used to record semistructured information in the Resource Description Framework being popularized by the World Wide Web consortium. The store uses standard elements (RDF, SOAP, XML, JDBC) to present a general purpose data model that can be used in a variety of components. We have begun discussion with other groups in the laboratory, for example the intelligent room project and the Start system, to explore using our store as a basic component of their systems. This will enhance the interoperability of several of the information management project in the laboratories.

With the databases layer prototyped, we dedicated much of the past six months to development of a suitable user interface. A challenge of our flexible data model is that little can be assumed about how information ought to be displayed: the display is quite dependent on what kind of information a user has placed in the system and how they want it presented. A raw view, simply listing the attributes of a particular object, is too complex and opaque to benefit a typical user.

To cope with this problem, we have developed a display mechanism that exploits the power of our general purpose data model. Objects in the data model have types; types have “display rules” describing, in RDF, how objects of that type ought to be displayed. The display rules are described in a semantic-web ontology. As a trivial example, one can specify that “document” objects (which tend to have an author and title) should be displayed with their titles in bold on top, and authors to the side. A user’s entire interface display can be described by a set of RDF assertions saying how collections of information (a “favorites” list, incoming email, recently accessed documents, a web home page) should be laid out on the screen. These two examples are common enough that we could imagine hardwiring them into a user interface. However, our approach gives a tremendous amount of flexibility. Should a new type of information become prevalent, one does not have to write a new “application” to deal with that type. Instead, some simple rules can be encoded in RDF, describing how that information should be displayed in a standard interface. These rules can be passed easily from information producer to information consumer. We expect this to be an important technique as we explore issues of collaboration (multiple users of interacting Haystacks) and translation of ontologies through our interaction with the Semantic Web project at the W3C.

Our previous report also discussed the development of the Winnow information spidering/filtering system. Over the past 6 month, we have carried out user test showing that the system lives up to its promised potential. The new machine learning algorithms developed for it provide excellent discrimination of information that will interest a given user. A paper discussing this research has been submitted.

Finally, we have spent the last six months carrying out a substantial user study. Our goal was to investigate the ways people work with their information, the intent being to use the results of this investigation to develop our Haystack user interface. The study consisted of 35 subjects, each of whom were monitored for a week and interviewed regarding their use of email, web browsing, filing of documents, and other information activities, and interviewed regarding what was observed. We are in the process of aggregating this information (transcribing interviews and classifying results) and are about to begin analyzing the resulting data.

Research Plan for the Next Six Months

With our initial implementations of a semistructured database and user interface complete, we have the basis of a usable system. One of our major goals in the next six months is to put the system to a test. We have mapped out a design by which Haystack can be used as a users primary email client. Email is a rich, dynamic source of information which is crying out for better organization. Despite the fact that a great deal of important information tends to be available only in an old email message, there are few tools available for searching through large email corpora. Many of us have had the experience of knowing that some email contains the information we want but being unable to find it. We believe Haystack can be used to improve this situation. Besides the obvious basic capabilities of full-text and attribute-based search through email, we expect to use Haystack’s machine learning features to

1. automatically classify email into appropriate folders,
2. filter out the most important and urgent email from less important material and spam
3. bring to a users attention and old email that may be relevant to one he is currently reading

4. make it easy for individuals to “recommend” email to other users, generalizing the idea of forwarding to incorporate measures of quality and rationales for the recommendation.

We intend to use email as the primary use-case, but expect that much of our work will also be relevant to assited web browsing and navigation of individual’s file systems.

To support several of these goals, we need to build up Haystack’s as-yet smallest pillar, machine learning and automated inference. We will implement algorithms for text clustering and text classification. We will also begin investigating ways to incorporate our non-textual semistructured data into the learning framework as well. Clearly, attributes such as sender and subject should play an important role in classifying email; this specific plan can be hardwired, but we would like to develop a framework that automatically generalizes to other data types than email.

Our development of the mail interface will be guided by the results of our user study. Over the next six months we expect to analyze the data we previously gathered from user interviews and use it to draw conclusions about the way to build an interface for information management.

Finally, we expect to begin exploring problems in collaboration. When multiple users, each with their own information ontologies, wish to collaborate, we will run into problems translating information from one user’s ontology to another’s. We have been working with the W3C on a “semantic web” project that investigates ways to surmount this translation barrier. We are only beginning to understand the key questions surrounding this problem, and hope in the next six months to begin producing some answers.