Paul B. Kantor, Endre Boros, Benjamin Melamed, Vladimir Meñkov, Bracha Shapira, and David J. Neu

Capturing HUMAN INTELLIGENCE in the Net

A new software system allows Web searchers to connect with others who have been there, done that.

What if your navigation of the Web were assisted not only by various search engines and software systems, but also by many other people who had searched for similar information in the recent past? This article discusses a software system called "AntWorld"[1] that integrates humans as part of the enabling technology to help other humans navigate the Web.

As a resource the Web is amazing and bewildering, and, at times, infuriating. All of us have, at one time or another, followed a seemingly endless loop, hopefully clicking one more time in a quest for some specific information. Many of us were also not the first person ever to be frustrated searching for that particular information. But the Web does not (yet) learn from other people's mistakes. In that sense, we who use it are not even as clever as ants in the kitchen, who always leave chemical trails for their nestmates when they find something good to eat.

Pursuing the ant metaphor, we imagine a user community operating in *asynchronous collaboration* mode, where information trails from user quests for information on the Internet are left behind for any community member to follow. The goal is to post and share communal knowledge: as community members engage in individual information quests, they make a small extra

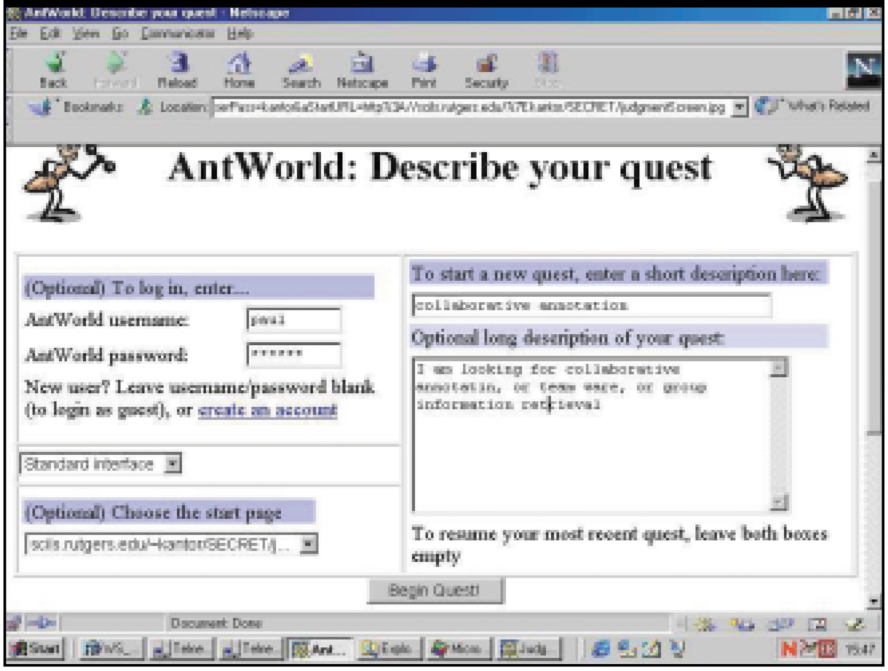
[1] aplab.rutgers.edu/ant/

JEAN-FRANCOIS PODEVIN

effort to contribute to the communal knowledge repository so that others may benefit.

Our aim is to introduce and facilitate this kind of asynchronous collaborative activity to increase the speed and accuracy of finding information on the Web. The software system must maintain the databases and perform the analyses needed to support user searches. As the user searches the Web, the system can then make page recommendations, either by proposing a list of relevant URLs or by visually marking recommended links with an icon. Recommendations must be based on extensive computation, which profiles quests, and summarizes them in short digital information pheromones (DIPs). Such profiling goes beyond customary profiling (as when consumers are profiled for interests). An effective profile of a quest should ultimately contain information not only about what the user is looking for, but also about the context and knowledge the user brings to bear. All of this is reflected, to some degree, in how pleased or displeased the user is with each particular page encountered in the search. This feedback is collected by having users judge the "goodness" of the pages encountered in a search.

Figure 1. The Quest Description window is presented at the start of a session, or when the user chooses to start a new quest. It provides for a user name and password at initial login. The text boxes accept short text (generally people enter keywords here), and a long description (note the typographical error; AntWorld does not correct such errors), which may be a natural language representation of the problem. Users may vary the interface setting, and/or choose a specific starting URL, before clicking the 'BEGIN QUEST' button to start. AntWorld begins by searching the base of stored queries for similar queries.



AntWorld: Describe your quest

(Optional) To log in, enter...

AntWorld username: pavai

AntWorld password: ******

New user? Leave username/password blank (to login as guest), or create an account

Standard interface

(Optional) Choose the start page

scils.rutgers.edu/~kantor/SECRET/j...

To start a new quest, enter a short description here:

collaborative annotation

Optional long description of your quest:

I am looking for collaborative annotatin, or team ware, or group information retrieval

To resume your most recent quest, leave both boxes empty

Begin Quest!

The AntWorld concept can be realized in three distinct architectures: *centralized, local proxy server,* and *distributed servers.* The centralized architecture is realized via a set of database files, all maintained at one physical location. The local proxy server can be hosted either on the user's client machine or as a network proxy server. In the long run, a distributed architecture could be hosted on servers distributed around the Internet and would support integration of local and global user databases of AntWorld information.

As Figure 1 illustrates, users provide explicit statements of their interests, both in terms of a few key words (short description) and in a natural language text (long description). The latter may serve either as an aid to memory, or as a source of additional terms. Note that these descriptions are not necessarily queries, although we do support the option of passing the key words direct to a selected search engine as a query. These descriptions provide short texts that define the notion of "similar quests" in the first search of the stored database of quests.

Key to the AntWorld concept is that individual searchers will expend the time and energy to provide judgments about the pages they visit during a search. The option JUDGE! is always available on the console. A user who selects this option reaches the judgment servlet or applet via an interface shown in Figure 2. Note the judgment button selections are converted to numerical scores, while the annotation text is preserved intact. Whether this annotation can be used in an automated search process is an open question, since we tend to use it for rather personal comments about the pages reached.

Inferred Judgments and Quests

There is a great deal of interest in inferring judgments of pages from automatically observable behavior of the user (that is, information that can be captured by the server or the browser, with no active effort by the user). Based on some preliminary work, we are skeptical about the value of the part of this information that is available to the server. On the other hand, we are finding that searchers tend to follow their eyes with the mouse (presumably in order to facilitate the necessary clicking when their eyes fall on an interesting option). Thus, information gleaned by recording and analyzing mouse activity might be of real value in trying to infer what the user is thinking.

Matching new quests to old ones is a form of the information retrieval problem. That is, the stored quest can be thought of as the documents in a digital library, and the current quest can be thought of as an evolving query against that library. We are exploring two main approaches to this issue.

Most current search and retrieval systems build on the so-called "Vector Space Model" [5] where a complex document is reduced to a string of numbers, representing the number of times that each different term (type, not token) appears in the document. Typically, grammatical variants of the same root may be counted as the same type, and in more complex schemes, related words may also be counted together. In advanced systems the notion of "term" may be enlarged to refer also to multiword phrases, proper names, dates, and other meaningful expressions. Once the query and each document have been represented by numbers, the similarity of a document to the quest can be defined as a "sum." This sum includes, for all the terms of the query, the product of the importance of the term in the query, its frequency in the document, and some overall metric factor based on the expected usefulness of that term in distinguishing among documents. This is often referred to as the *tf.idf* approach (for "term frequency by inverse document frequency"). More formally, let $f(t,x)$ represent the frequency of term t in the entity x, which may be a query (q) or document (d), and let the expression $w(t,t')$ to represent the importance of term types, as well as their interrelation. Then the similarity of a query to a document $Sim(q,d)$ can be written as

$$Sim(qd) = \sum_{terms\ t} f(t,q)f(t,d')w(t,t')$$

The Vector Space Model works rather well. It can be shown to follow rigorously from deeper Bayesian assumptions about the distribution of terms which, somewhat paradoxically, are known to not hold very accurately for spaces of real documents and queries. Thus, we regard this formula as a useful phenomenological description of the concept of similarity among texts. Our AntWorld system implements a number of vector space-based methods, as discussed in [4]. Because AntWorld is still a research tool, methods of computation can be varied dynamically, as part of a system configuration.

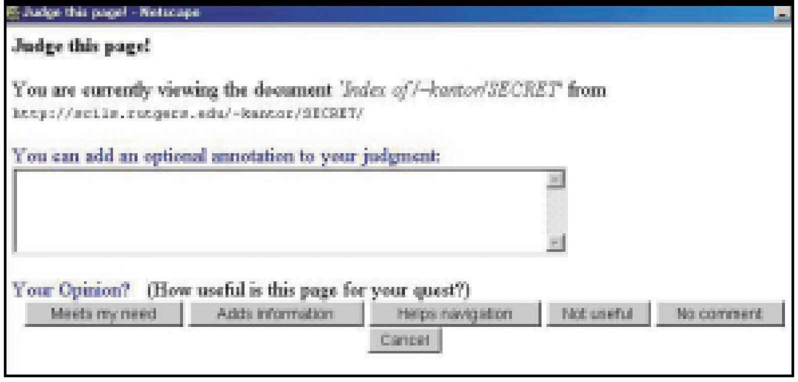
KEY TO THE ANTWORLD CONCEPT IS THAT INDIVIDUAL SEARCHERS WILL EXPEND THE TIME AND ENERGY TO PROVIDE JUDGMENTS ABOUT THE PAGES THEY VISIT DURING A SEARCH.

The TREC[2] experience has shown that carefully built handmade weighted Boolean queries can be more effective in information retrieval than even the best of the methods based on term frequencies and the Vector Space Model. However, it is also known that making use of judgments (often called "obtaining relevance feedback") greatly improves the performance of a retrieval system. There are no publicly described systems in which these two strengths are combined. We believe this stems from the difficulty of automatically building a Boolean representation, given the user's relevance feedback. A key technical issue in the development of AntWorld is our attempt to unite these two features, building on a theory called "Logical Analysis of Data" (LAD), developed by Peter Hammer and colleagues at Rutgers University. LAD takes the perspective that the user has rules for deciding how to judge a page, and that these rules form an incompletely defined Boolean function. To apply LAD to the relevance feedback problem we treat the user's relevance judgments as an incompletely defined Boolean function.

A Boolean function involves any number of variables whose values and arguments are binary (for example, taking values 0 or 1). It is convenient to represent each of the possible user judgments ("meet my need," "adds information," "not relevant") by a separate Boolean function. For each of these functions,

[2] trec.nist.gov

Figure 2. The Judgment window. The user may dismiss this window by selecting any of the buttons. Optionally, text can be entered in the annotations box. This is likely to be an annotation that helps the user and/or colleagues to understand this page in the context of the Quest as a whole. AntWorld does not attempt to process this automatically.



Judge this page! - Netscape

Judge this page!

You are currently viewing the document *'Index of /~kantor/SECRET'* from
http://scils.rutgers.edu/~kantor/SECRET/

You can add an optional annotation to your judgment:

Your Opinion? (How useful is this page for your quest?)

LAD then builds a table of all the known cases. These cases include positive examples (the function returns the value 1), and negative examples (the function returns the value 0). Building on the Vector Space Model, each of the examples is also represented by the vector whose elements are the frequencies in which terms occur. Our second challenge is to binarize this vector in a sensible way. We do this in a greedy fashion, finding the best cutoff frequency for each term, so that documents with frequencies above the cutoff tend to be positive cases. Those with frequencies below the cutoff tend to be negative cases. We then define a binary variable x_{term} to be 1 if the frequency of the term is above cutoff, and 0 if it is below the cutoff. After these steps are completed we have a set of tables giving the value of the function for various values of its arguments. We then invoke the LAD machinery to search for the most economical and effective Boolean rule that reproduces the observed table of positive and negative cases.

In actual use, the resulting Boolean rule is taken to represent the user's quest, and that rule is applied to all the stored quests to select those most similar to the user's quest. In practice, we need more than a simple YES/NO judgment about older stored quests, so we have developed an ad hoc method for assigning weights to documents based on how strongly they match individual terms in the Boolean function.

The Future

AntWorld can run at any number of sites simultaneously, and the databases of judgments/quests can evolve together as common interests emerge. Obvious starting points are help desk services that deal with a base of logged queries. Another starting point is in military applications, where it may be necessary to research a topic (for example, a geographic, or political situation) very quickly, and where it is important to draw upon all the relevant earlier searches. Competitive intelligence is another area of business application. AntWorld could also serve within a large corporation, helping to control the mass of machine-readable reports and memoranda in a new, effective fashion using people to provide judgments/quests.

In our estimation, these topic-specific and organization-specific models represent the most likely starting points for the evolution of AntWorld. The eventual coevolution of these distributed ant colonies might follow the same path as the evolution of business-to-business online markets. That is, sites and organizations with similar interests would find it useful to combine their operations, realizing economies of scale and network externalities as the system grows. C

References

1. Boros, E., Kantor, P.B., and Neu, D.J. Pheromonic representation of user quests by digital structures. In *Proceedings of ASIS'99*; aplab.rutgers.edu/ant/papers/
2. Boros, E., Horiyama, T., Ibaraki, T., Makino, K., and Yagiura, Y. Finding small sets of essential attributes in binary data. DIMACS Technical Report DTR 2000-10. Rutgers University; ftp://dimacs.rutgers.edu/pub/dimacs/TechnicalReports/TechReports/2000/2000-10.ps.gz.
3. Boros, E., Melamed B., and Meòkov, V. The information quest: A dynamic model of user's information needs. In *Proceedings of ASIS'99*; aplab.rutgers.edu/ant/papers/
4. Meòkov, D. Neu, J., Shi. Q. AntWorld: A collaborative Web search tool. To appear in *Proceedings of the Third Workshop on Distributed Communities on the Web (DCW 2000)*; aplab.rutgers.edu/ant/papers.
5. Salton, G., McGill. M.J. *Introduction to Modern Information Retrieval.* McGraw-Hill, NY.
6. Shapira, B., Kantor, P.B., and Melamed, B. Preliminary study of the effect of motivation on user behavior in a collaborative information finding system. Technical Report; www.business.rutgers.edu/~shapira/paper2.htm

Paul B. Kantor (kantor@scils.rutgers.edu) is a professor of information science in the School of Communication Information and Library Studies; **Endre Boros** (boros@rutcor.rutgers.edu) is a professor of operations research at RUTCOR; **Benjamin Melamed** (melamed@rbs.rutgers.edu) is a professor of management science and information systems at the Faculty of Management; **Vladimir Meñkov** (vmenkov@cs.indiana.edu) was a research associate at the Alexandria Project Laboratory; and **Bracha Shapira** (shapira@business.rutgers.edu) is a lecturer (on leave) at the Department of Management Science and Information Systems—all at Rutgers University, New Brunswick, NJ. **David J. Neu** (neu@rutcor.srutgers.org) is a Ph.D. candidate at the Rutgers Center for Operations Research and is a chief scientist with UTSR, Inc., in Cherry Hill, NJ.

This project was supported by DARPA (Contract Number N66001-97-C-8537), and the Rutgers University Alexandria Project Lab and Distributed Laboratory for Digital Libraries.