Beyond Similarity

Jay Budzik, Kristian J. Hammond, Larry Birnbaum, and Marko Krema

Intelligent Information Laboratory
Northwestern University
1890 Maple Ave.
Evanston, IL USA
{budzik, hammond, birnbaum, krema}@infolab.nwu.edu

Abstract

Agents that provide just-in-time access to relevant online material by observing user behavior in everyday applications have been the focus of much research, both in our lab, and elsewhere. These systems analyze information objects the user is manipulating in order to recommend additional information. Designers of such systems typically make the assumption that objects *similar* to the one being manipulated by the user will be useful to her. Our own experiments show that users do find many of the documents retrieved by a system of this type are *relevant*. Yet in the context of a specific task, users find fewer of these documents are *useful*.

Our main point is that in order to make just-in-time information systems truly useful, we need to reexamine the "similarity assumption" inherent in many of these systems' designs. In light of this, we propose techniques that bring modest amounts of task-specific knowledge to bear in order to perform lexical transformations on the queries these systems perform, thereby ensuring they retrieve not similar documents, but documents that are relevant and useful in purposeful and interesting ways.

Introduction

Just-in-time information agents are systems that observe user behavior in everyday applications (e.g., word processors, WWW browsers, electronic mail systems), and build queries to distributed information repositories in order to provide a user with immediate access to relevant information. Efforts in building such systems have varied in the kind of document collections and applications used, the level of user modeling involved, the amount of user intervention, and the interface for presentation (Lieberman 1995, Rhodes and Starner 1996, Badue, Vaz, and Albuquerque 1998, Budzik et al. 1998, Kulyukin 1999, Budzik and Hammond 2000, Rhodes 2000, Maglio 2000).

Designers of such systems typically make the assumption that the goal of the system should be to retrieve objects that are similar to the one currently being manipulated (e.g., to "find more like this" (Turney 1999)). This is motivated by the underlying vector-space model of information retrieval (Salton, Wong, and Yang 1971), typically used by such systems. In the vector-space model, requests are matched against information objects by measuring the similarity of the request and objects in a database. Documents and queries are represented as vectors of term weights in a high dimensional space (the order of the space is determined by the number of unique words or word stems in the corpus). Given a query Q, a vector in this space, documents D for which $d(Q, D)$ is minimized are retrieved, where d is some measure of distance (usually the cosine of the angle between the vectors Q and D, or equivalently the dot product of the two vectors if the space is normalized).

This basic model is used by many information retrieval systems, including most Internet search engines. Just-in-time information agents typically build queries to these systems in order to recommend related documents. The goal of the underlying information retrieval system used by these agents is to match a request with the most similar document. It is likewise often the goal of the agent to find documents that are similar to the one the user is currently manipulating.

We have come to take a very different position: that the goal of just-in-time information agents should not be to recommend similar information objects, but instead to find objects that are useful to the user in the context of the task she is performing. In many cases, this involves discovering information objects that are quite different from the document at hand.

The following sections describe previous work on just-in-time information agents that has motivated this position. We go on to describe early work on systems that use techniques that bring modest amounts of task-specific knowledge to bear in order to perform lexical transformations on the queries these systems perform. The goal of the techniques we describe is to ensure the system retrieves not similar documents, but documents that are relevant and useful in purposeful and interesting ways.

Just-in-time Information Systems

The development of just-in-time information agents stems from the observation that information repositories are accessed by users from the same machine on which they

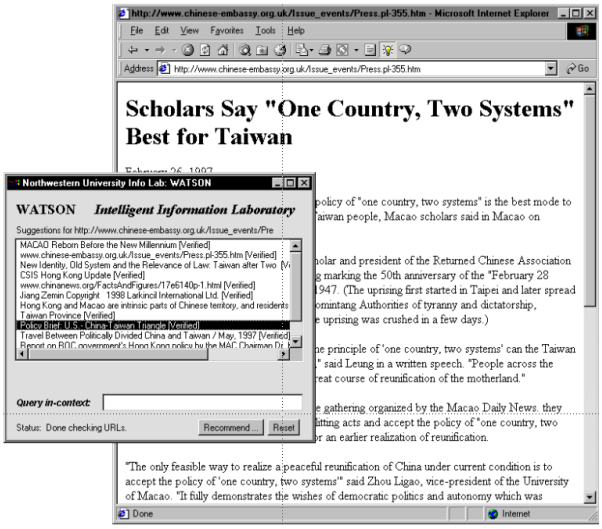


Figure 1: Watson is recommending related documents to a user reading a Web page.

write papers, read news, and browse the Web sites that interest them. The aim of research on just-in-time information agents is to expand the bandwidth of information available to information systems by coupling them tightly with the tasks the user is performing in other applications. Thus requests for information need not be construed in an isolated setting, devoid of external influence. Rather, information requests can be grounded in the context of the activities the user is performing, and results can be judged by their utility in this context.

Our work on Information Management Assistants (Budzik et al. 1998, Budzik and Hammond 2000) has focused on modeling user behavior in everyday productivity applications to make decisions about when to retrieve relevant information, and what kind of information would be useful to the task at hand. We have built a prototype system, Watson, which observes user behavior in Microsoft Word (a word processor) and Microsoft Internet Explorer (a WWW browser) in order to decide when and from where to retrieve related information. The system then analyzes the content of the document the user is manipulating in order to build queries to distributed Internet and Intranet information repositories (e.g., search engines like AltaVista (AltaVista 1995)).

The exact details of the algorithms Watson uses for query generation are described elsewhere (see (Budzik and Hammond 2000)). Briefly, the system analyzes documents using a heuristic term weighting algorithm that favors words indicative of content, based on their frequency within the document and their presentation attributes (e.g., if a word is in a large font, it receives a higher weight; if it is smaller than the rest of the text or embedded in a list of links, it receives a smaller weight).

Results from the automated searches Watson performs are analyzed to detect duplicates, and presented in a separate, background window (see Figure 1). The user can then click on a list entry in order to view the corresponding Web page in a separate window. Users can also direct explicit queries to the system, which are processed in the context of the document the user is currently manipulating. Watson combines the terms from a user's explicit query with the terms it has used to retrieve related documents. For example, if a user is browsing a page on Mars, and enters the query "life" into Watson's query box, Watson will return pages about life on Mars. We call this facility *query in context*. The following scenario illustrates how the system might be used.

Usage Scenario

Suppose, for example, our user is doing research on the relationship between Taiwan and China. She finds a document from Chinese Embassy, stating China's position. She notes this document's extreme bias on the side of reunification, and would like to find other documents that take a different perspective. The Chinese Embassy pages are of no help to her; they have no links to external sources. She goes to her Watson window. Watson has been watching her browse, and has gathered a list of related documents (see Figure 1). One particular document catches her eye: a policy article from the US embassy about Taiwan and China. The US document has a distinctly different perspective—it is more concerned with the economic impact of the reunification for the United States. Yet another document explains how groups in Taiwan are viewing this issue.

This example illustrates that similarity on a topical level plays an important role in the success of such systems, but that ultimately, it is the user's goals in the context of a particular task that determine whether the results are useful. Similarity alone cannot always account for this. In every case, the most similar document is always the same document. In many cases, documents representing the same or similar points of view are redundant and useless. The following section describes empirical work we have done that supports this view.

Experiments

Most systems that make recommendations are evaluated on the basis of the objective relevance of a recommendation given the input to the system (e.g., (Badue, Vaz, and Albuquerque 1998, Rhodes 2000, Kulyukin 1999, Jansen and Spink 1998)). Other systems are evaluated based on some measure of how "good" the recommendation is, a main component of which is usually relevance (e.g., (Dean and Henzinger 1999)). Our first experiment was modeled after this general methodology.

Experiment 1

For our first study, we collected a list of Web pages from other researchers at Northwestern. We asked users to choose a page from the list, look at it in a Web browser and then use Alta Vista to find similar pages. The users then

	# Useful	# Similar	# Returned	% Useful	% Similar	Difference
Subject 1	4	8	18	0.22	0.44	0.22
Subject 2	1	9	11	0.09	0.82	0.73
Subject 3	0	1	8	0.00	0.13	0.13
Subject 4	8	8	13	0.62	0.62	0.00
Subject 5	2	6	9	0.22	0.67	0.44
Subject 6	5	3	14	0.36	0.21	-0.14
Mean				0.25 ± 0.22	0.48 ± 0.27	0.23 ± 0.24

Table 1: Users judged pages returned by Watson by their utility in the context of their task and by their similarity to the document they were manipulating on a 5-point scale. Judgments $n > 2$ were considered similar or useful.

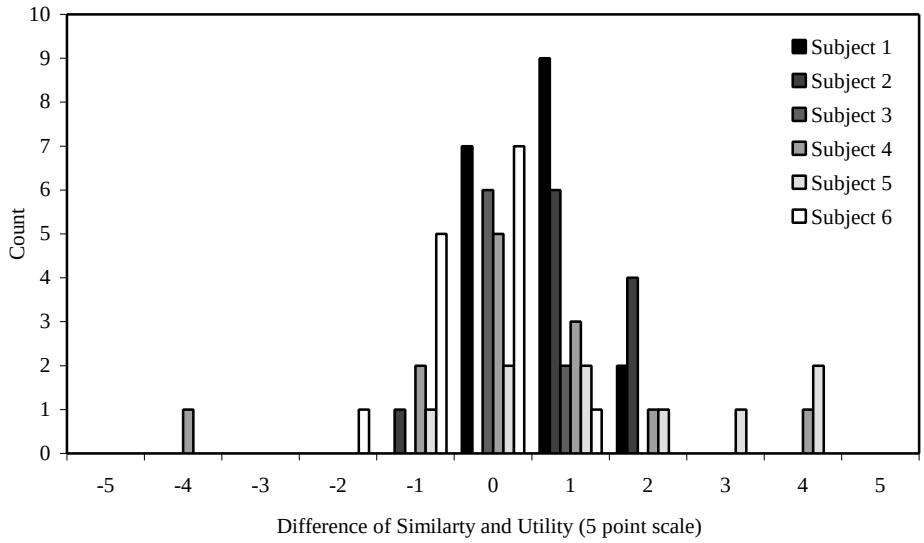


Figure 2: Histogram of the difference between similarity and utility (on a 5-point scale) for 6 subjects. The overall mean is 0.53, with standard deviation 1.29. The means and standard deviations of the individual subject rankings are shown in Table 2, below.

	Mean Difference	Std. Deviation	Correlation
Subject 1	0.72	0.67	0.85
Subject 2	1.18	0.87	0.35
Subject 3	0.25	0.46	0.82
Subject 4	0.23	1.83	0.29
Subject 5	1.56	1.81	0.11
Subject 6	-0.43	0.76	0.80

Table 2: Mean difference between similarity and utility, standard deviation of the mean, and correlation between similarity and utility for 6 subjects. The overall mean is 0.53, with standard deviation 1.29. The overall correlation is 0.51.

judged the top 10 pages returned as relevant or irrelevant to their search task. Next, the users were asked to judge the sites Watson returned from the same page in the same way. In this experiment, Watson used Alta Vista as well. For our initial group of subjects, we drew from local computer science graduate students. All of the volunteers considered themselves expert-level searchers. This was evident in their query behavior, as most of them used long queries (≥4 words), laden with advanced features.

We gathered 19 samples from a pool of 6 users. Using Alta Vista, our group of expert searchers was able to pose queries that returned, on average, 3 relevant documents out of 10. Watson was able to do considerably better at the same task, returning, on average, 5 relevant documents out of 10. In the samples gathered, Watson was able to do as well or better than an expert user 15 out of 19 times.

While the results of the experiment were favorable, users complained that they did not know on what basis they should judge relevance. Moreover, it was unclear that the similar documents were appropriate for some pages. For instance, one of the test pages was the front page of Yahoo! (Yahoo! 1994). It is unlikely there is any page that can objectively be considered relevant to this Internet directory site, in the absence of a knowledge of the user's specific goal.

The above observations caused us to question the methodology used to evaluate the system, as well as the goal of the system, in general. Thus while this experiment was aimed at judging the *relevance* of Watson's recommendations given a document, the second experiment attempted to measure the *utility* of results within a particular task context. The results were remarkably different.

Experiment 2

This experiment was aimed at determining whether or not the sources returned by Watson were useful in the context of a particular task. Because Watson is intended to work alongside the user as she is completing a task, we were convinced that evaluating the utility of the information provided was more appropriate than the relevance-based judgments that are typical of most other evaluations of information retrieval systems and recommender agents. In addition, we were interested in investigating the degree to which the similarity of a returned document was related to the utility of that document in the context of a task.

We asked researchers in the Computer Science department to submit an electronic version of the last paper they wrote. Six responded. Each paper was loaded into Microsoft Word while Watson was running. The results Watson returned were then sent to the authors of the paper. Watson returned 74 documents for all of the respondents (on average, Watson returned 12 documents per subject). Subjects were asked to judge the degree to which a recommended document would have been useful to them in the context of the task they were performing. Subjects were also asked to judge how similar the retrieved document was to the document they gave us. Both judgments were recorded on a 5-point scale.

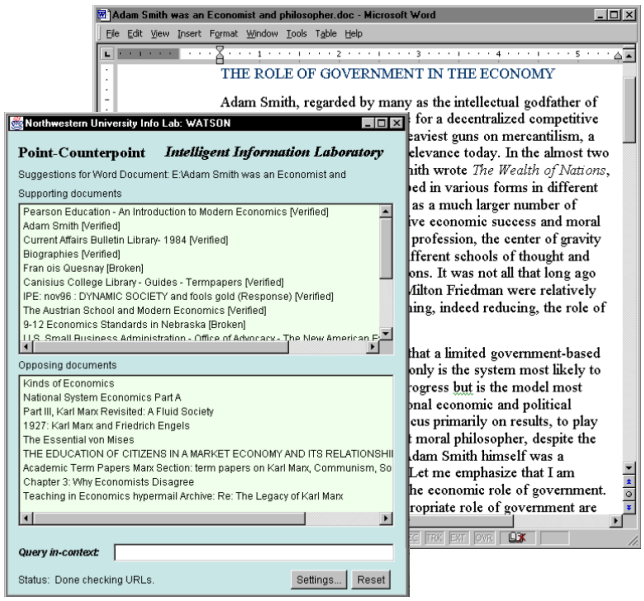
For the following summary statistics, a document was counted as similar or useful if subjects gave it a numeric ranking above 2. All of the subjects indicated that at least one of the references returned would have been useful to them. Two of the subjects indicated the references Watson provided were completely novel to them, and would be cited or used in their future work. On average, 2.5 of the documents Watson returned were deemed useful in the context of the task the user was performing. In contrast, almost half of the documents Watson returned were judged similar (loosely replicating the results gathered in Experiment 1). Unlike in Experiment 1, subjects reported it was easy for them to make judgments. The results of this experiment are summarized in Table 1.

Figure 2 displays a histogram of the difference between similarity and utility rankings. The distribution is skewed to the positive end, indicating documents were often more similar than useful. In addition, the histogram indicates several documents were useful, but not similar. Conversely, the histogram indicates several documents were similar, but not useful. In fact, the correlation between similarity and utility was $r = 0.51$ (if $|r| = 1$, similarity and utility would be perfectly correlated, if $r = 0$, they would not be related at all). Thus the similarity of a result accounts for $r^2 = 0.26$, or about a quarter of the variance in the utility of a result. This weak overall relationship between similarity and utility underscores the necessity of evaluating the performance of these systems within the context of a particular task.

We also examined whether or not similarity was good enough for some tasks, given that it does not seem to be sufficient for all of them. Indeed, for subject 4 in our study, every similar document was also useful. The correlation of similarity and utility within a particular task context is presented in Table 2. The data in table suggest that for some tasks, similarity is indeed a good predictor of utility (correlations close to 1), but for others, they have little to do with each other (correlations closer to 0).

Discussion

The results of the above experiment show that for some tasks, *similar* documents are not *useful* documents. In the aggregate Watson did consistently produce useful documents for users. However, the results suggest that improvements aimed at addressing particular tasks would be worth making.



(a)

```
NAME: capitalism/communism
TERMS: capitalism
TERMS: wealth of nations, adam smith
TERMS: adam smith
TERMS: communism, economy
TERMS: marx, communism
TERMS: marxism, economy
TERMS: karl marx
TERMS: communist manifesto
RULE(ruleset1): SUBST karl marx/adam smith
RULE(ruleset1): SUBST adam smith/karl marx
RULE(ruleset1): SUBST marx/smith
RULE(ruleset1): SUBST smith/marx
RULE(ruleset2): SUBST capitalism/communism
RULE(ruleset2): SUBST communism/capitalism
```

(b)

Figure 3: Section (a) shows the Point/Counterpoint interface; Section (b) shows part of a sample issue file.

It is important to note that the above experiment only evaluated the automated results Watson returns. It did not evaluate the utility of results generated by explicit requests using Watson's query in context facility. Our hypothesis is that use of this facility would improve the results described above, but further empirical work is needed to conclude this is in fact the case.

The following section describes techniques aimed at improving the quality of results returned for a specific kind of document composition task.

Getting Beyond Similarity

We are in the process of developing a system that supports users in a specific class document composition tasks: writing opinion pieces. The system, Point/Counterpoint, is built on top of the Watson system. Point/Counterpoint assists users in supporting their point of view while they are developing a written argument.

The system is based on the idea that when formulating an argument in support of a particular point, other documents which represent arguments both for *and against* that point are useful references. Point/Counterpoint uses knowledge of opposing experts in particular domains to recognize opportunities to retrieve examples of contrary points of view. For example, when a user cites Marx's idea of an ideal economic state, Point/Counterpoint will retrieve two sets of articles: one set representing Marx's point of view, and another set representing Adam Smith's opinion (see Figure 3a).

Point/Counterpoint forms two separate queries—a *similar query* and an *opposite query*. The opposite query is formed by modifying the result of the original query generated by the Watson system, using substitution rules that essentially replace the name of an expert with his opposite while retaining the terms that represent the general topic of the argument. The similar query is an unmodified copy of the original query Watson generated.

Issues are represented in multiple issue files, which define an issue detector and transformations to be performed on queries. Figure 3b shows part of a sample issue file. Each issue file has a name, a set of term conjuncts used to detect the issue, and a set of rules used to perform lexical transformations on the queries the system sends to Internet search engines. Issues are activated by detecting disjunctions of term or phrase conjuncts in the document the user is manipulating. Each TERMS line in an issue file is treated as a conjunction of the listed terms or phrases. An issue is activated when one of these conjunctions is satisfied by the contents of the current document. For example, the issue defined in Figure 3b would become active if both of the phrases "adam smith" and "wealth of nations" were detected in the document being edited. The rules defined in the file are then applied to the original "find more like this" query generated by the Watson system[1], resulting in an *opposite query*.

Point/Counterpoint currently supports substitution rules only. The antecedent of a substitution rule is the term or phrase to be substituted. The consequent is the term or phrase with which the phrase or term matched in the antecedent will be substituted. For example, the first rule in Figure 3b specifies the phrase "karl marx" should be substituted with the phrase "adam smith." Each rule belongs to a rule set. Only one rule from each rule set is executed.

Multiple issue files may be active at the same time. If there are *n* active issues, *n* opposite queries are generated.

[1] Watson generates queries in all lower case.

The resulting queries are then executed by sending them to Internet search engines.

Currently, issue files are constructed manually. We are in the process of building crawling agents that automatically learn issues by extracting patterns of referral indicative of opposing points of view (e.g., "unlike ?x, ?y believes ?z") from collections of Web documents and research papers. This approach is motivated systems like Rosetta (Bradshaw and Hammond 2000) and Spin Doctor (Sack 1995) that use reference patterns for similar purposes.

We are also investigating approaches that leverage explicit user feedback across multiple users in similar tasks. Techniques in this vein are inspired by collaborative filtering systems (Goldberg et al. 1992, Konstan et al. 1997, Shardanand and Maes 1995), which recommend items to users by clustering groups of users with similar taste. If two users have liked many of the same items in the past, then the system can use the judgments of one user on a new item to predict the second user's judgment on the same item. Predictions of positive ratings can them be used to make recommendations. We hypothesize that in the context of composing a document, it is likely that users manipulating very similar documents will find many of the same recommendations useful in accomplishing their task.

Conclusion

In summary, we discussed a class of systems called just-in-time information agents. These systems analyze information objects the user is manipulating in order to recommend additional information. Designers of such systems typically make the assumption that objects similar to the one being manipulated by the user will be useful to her. Our main point is that this assumption is not a valid one—that in order to make just-in-time information systems truly useful, designers need to focus on ensuring results will be useful in the context of a particular task.

We presented an experiment that showed users do find many of the documents retrieved by a system of this type are *relevant*. A second experiment showed that in the context of a specific task, users find fewer of the recommend documents are *useful*. Moreover, it showed that overall, similarity was only a fair predictor of utility, yet for some tasks it was better suited than for others.

In light of this, we described ongoing work on a system called Point/Counterpoint that uses knowledge of opposing experts to perform lexical transformations on the queries the system performs. The aim of work on Point/Counterpoint is to ensure the system retrieves not similar documents, but documents that are relevant and useful in purposeful and interesting ways.

References

AltaVista (1995). Available at http://www.altavista.com/.

Badue, C., Vaz, W., and Albuquerque, E. (1998). Using an Automatic Retrieval System in the Web to Assist Co-operative Learning. 1998 World Conference of the WWW, Internet and Intranet, Orlando, FL, USA, AACE Press.

Bradshaw, S., Scheinkman, A., and Hammond, K. J. (2000). Guiding People to Information: Providing an Interface to a Digital Library Using Reference as a Basis for Indexing. 2000 International Conference on Intelligent User Interfaces, New Orleans, Louisiana, USA, ACM Press.

Budzik, J., Hammond, K. J., Marlow, C., and Scheinkman, A. (1998). Anticipating Information Needs: Everyday Applications as Interfaces to Internet Information Sources. 1998 World Conference of the WWW, Internet and Intranet, Orlando, FL, AACE Press.

Budzik, J., and Hammond, K. J. (2000). User Interactions with Everyday Applications as Context for Just-in-time Information Access. 2000 International Conference on Intelligent User Interfaces, New Orleans, Louisiana, USA, ACM Press.

Dean, J., and Henzinger, M. R. (1999). Finding related pages in the World Wide Web. Eighth International World Wide Web Conference, Elsevier.

Goldberg, D., Nichols, D., Oki, B. M., and Terry, D. (1992). "Using collaborative filtering to weave an information tapestry." Communications of the ACM 35(12): 61-69.

Jansen, B., Spink, A., and Bateman, J. (1998). Searchers, the Subjects they Search, and Sufficiency: A Study of a Large Sample of EXCITE Searches. World Conference of the WWW, Internet and Intranet, AACE Press.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. (1997). "Applying collaborative filtering to Usenet news." Communications of the ACM 40(3): 77-87.

Kulyukin, V. (1999). Application-Embedded Retrieval from Distributed Free-Text Collections. The Sixteenth National Conference on Artificial Intelligence, AAAI Press.

Maglio, P., Barrett, R., Campbell, C., and Selker, T. (2000). SUITOR: An Attentive Information System. 2000 International Conference on Intelligent User Interfaces, New Orleans, Louisiana, USA, ACM Press.

Rhodes, B. (2000). Margin Notes: Building a Contextually Aware Associative Memory. 2000 International Conference on Intelligent User Interfaces, New Orleans, Louisiana, USA, ACM Press.

Sack, W. (1995). Representing and Recognizing Point of View. The American Association of Artificial Intelligence Fall Symposium on Artificial Intelligence Applications in Knowledge Navigation and Retrieval, Cambridge, MA., AAAI Press.

Salton, G., Wong, A., and Yang, C. S. (1971). "A vector space model for automatic indexing." Communications of the ACM 18(11): 613-620.

Shardanand, U., and Maes, P. (1995). Social Information Filtering: Algorithms for Automating 'Word of Mouth'. CHI-95, The 1995 International Conference on Human Factors in Computing, Denver, CO, ACM Press.

Turney, P. (1999). Learning to Extract Keyphrases from Text, Tech. Report Number NRC-41622, National Research Council Canada, Institute for Information Technology.

Yahoo! (1994). Available at http://www.yahoo.com/.