KAVANAH: AN ACTIVE USER INTERFACE INFORMATION RETRIEVAL APPLICATION

EUGENE SANTOS JR. AND HIEN NGUYEN
Intelligent Distributed Information System Lab
Department of Computer Science and Engineering
University of Connecticut
{eugene,hien}@cse.uconn.edu

SCOTT M. BROWN
Air Force Research Laboratory
Crew System Interface Division
Wright-Patterson AFB, OH
sbrown777@acm.org

This paper reports our implementation and evaluation of an active user interface in an information retrieval application called Kavanah. The goal of the active user interface is to improve the quality of information retrieval and to reduce the user's cognitive workload while searching for information. Our underlying concept is to dynamically construct the search queries based on a dynamic representation that captures user interests, preferences and searching context (as represented in a user ontology). Our approach to disaggregating the essential aspects of a user's intent for searching allows for focused multi-agent based construction and correction of the overall user model that captures the user's intent, thus promoting increased effectiveness and efficiency. We evaluate the effectiveness of the active user interface with commonly used metrics from the information retrieval community by measuring retrieval performance with and without the presence of an active user interface. Furthermore, we measure the ability to discover new knowledge by evaluating our dynamic online ontology construction. The evaluations use the Unified Medical Language System knowledge base as a test bed.

1 Introduction

During the last few years, as the result of the overwhelming number of choices of online and offline information resources, we have witnessed an increasing trend towards the construction of personal assistant agents in information filtering, recommender systems and agent communities[2,9,11]. The main focus of these approaches is to capture user interests by analyzing the user interactions with the system and to use these interactions to guide the system reactions accordingly to improve the quality of the users' work.

In this paper, we hypothesize that constructing a unified model of the user's interests, preferences, and context in an information seeking task provides a fine-grained model that more effectively captures the *user's informa-*

tion seeking intent than a model addressing a subset of these salient characteristics. While other previous efforts have focused exclusively on learning any one aspect of information seeking, none of them has attempted to integrate all three aspects together for determining a user's intent in seeking information. We refer to our personal assistant agent as an active user interface (AUI) in this paper. Active user interfaces not only capture user interests, preferences, and contexts but also focus on the interactions among them in a dynamic fashion. In particular, our focus is on deriving and learning the context or user ontology. Most existing methods assume that all users share a single common ontology[13]. This implicitly assumes that all users have the same level of understanding and beliefs expressed in the common ontology. We believe that users understand information and how it interacts in their own individual way. This arises from many factors ranging from user experience and expertise to basic differences in user style and operation. We show that by using our model, we can do more than just elicit the user interests and preferences. We provide a learning capability for the system to discover new knowledge based on analyzing the documents relevant to the user and the context, i.e. *why* the user is focusing on the given information. This work is derived from our earlier research with a predecessor system, Clavin[4,15,16].

We evaluate our hypothesis by constructing an AUI in an information retrieval application called Kavanah. The implementation of our AUI is a multi-agent based system in which the main agent contains the user model consisting of user preference, interest, and context and the suppporting agents are used to dynamically construct and maintain the user model based on changes in the user's intent as well as incorrectness and incompleteness in the user model. Our evaluation goal is to show the effectiveness of this model by comparing the system performance in cases with and without an AUI using commonly used metrics in information retrieval.

The rest of the paper is organized as follows: the next section discusses the architecture of the system followed by a detailed description of our implementation. Next, we discuss our preliminary empirical evaluation. Finally, related work and future research issues are considered.

2 System architecture

The main goal of Kavanah is to use its AUI to assist the users in getting the right information at the right time using the right tools[4]. The goal of the AUI is to accurately represent a user's intent. Intent inference involves deducing an individual's goals based on observations of that individual's actions[12]. In automated intent inference, this process is typically implemented through one or more behavioral models that have been constructed and optimized for the individual's behavior patterns. In an automated intent inference system, data representing observations of an individual, the individual's actions, or the individual's environment (collectively called *observables*) are collected and delivered to the model(s), which match the observables against patterns of behavior and derive inferred intent from those patterns. These inferences can then be passed to an application for generation of advice, definition of future information requirements, or proactive aiding.

We partition intent inference into three formative components. The first, *interests*, captures at a high level the focus and direction of the individual's attention. The second, *preferences*, describes the actions and activities that can be used to carry out the goals that currently hold the individual's attention, with a focus on how the individual tends to carry them out. The third, *context*, provides insight into the user's knowledge and deeper motivations behind the goals upon which the individual is focused and illuminates connections between goals. In other words, the first component captures *what* the individual is doing, the second captures *how* the individual might do it, and the third infers *why* the individual is doing it. With regards to the research presented in this paper, the AUI needs to provide the right assistance to the information retrieval application on *what* the user is currently interested in; *how* a query needs to be constructed and returned results needs to be portrayed; and *why* the user dwells on a search topic.

We assume that the interests are influenced by the ultimate goal that the user is trying to reach and the methods which she uses to accomplish that goal. For example, suppose that the user's goal is to study lung cancer and her approach is to scan materials from general definitions to specific methods used to treat this disease. Her interests will thus vary from general treatments to specific chematography processes. In particular, her interests may change from a certain drug to a more general approach for treatment. The user interests, in turn, influence user preferences and context. If user interests appear to be far off the goal that the user is trying to reach, she may change her search strategies and understanding of the subject accordingly.

In our AUI, we capture the interest, preference, and context aspects of user intent with an *interest relevancy set*, a *user ontology network*, and a *preference network* correspondingly. The *interests relevancy set* determines what is currently relevant to the user. It is generated by reasoning over the user ontology network. Based on the utility values of each concept node in the user ontology network, we end up with a rank ordering of the concepts to build an interest relevancy set. Since user interests change over time, we incorporate a fading function to make the irrelevant interests fade away. We will describe this process in more detail in Section 3.

The *user ontology network* captures the user's knowledge of concepts and the relations among concepts in a specific domain. Before further discussing the user ontology network, we introduce briefly the concept of a domain ontology. The domain ontology captures the domain knowledge containing the concepts and the relations among them in a specific domain. The user ontology exploits the domain ontology by extracting the missing information that it needs to have in order to help identify the concepts which the user is interested in. Therefore, the user ontology will be similar to a subgraph of the domain ontology. However, in the user ontology network, additional relations which are not found in the domain ontology may exist as a result of user misconceptions or user-specific expert knowledge that is not fully captured in the domain ontology. The user ontology network in Kavanah is represented by a Bayesian network in which each node either represents a concept or a relation among concepts.

The *preference network* represents *how* the user wants to form the query, how this query should be answered, and how results should be portrayed. The user's preference in Kavanah is reflected by how the user prefers to use a class of *tools*. A tool is defined as an operator to perform specific actions to transform the information that the user needs based on preference. Each node in the preference network represents a tool, an action associated with that specific tool, or a pre-condition which represents the requirements of the tool connected with it. An example of a tool is a filter that removes those documents that do not match a certain criteria. Another example of a tool is an expander that searches for documents that expand the searching topic. Figure 3(a) shows an example of a preference network.

The AUI uses correction adaptation agents to maintain the preference network. Each correction adaptation agent offers a bid to the AUI to change the preference network. They maintain a user model that is identical to that of the AUI until the AUI requests bids, at which time the AUI adapts its own user model based on its bidding behavior component. In the bidding process, the correction adaptation agent that most likely improves the AUI's effectiveness will win the bid and this winning agent is permitted to correct the user model. We evaluate the user model by a set of metrics that measure its adaptivity, autonomy, collaboration and robustness requirements. We capture the user's utility for having the AUI perform an action on his behalf to achieve a goal by the utility function over that set of requirements. An example of such a utility function is as follows: $U_{req} = 0.14 * U_{reactive} + 0.14 * U_{predictive} + 0.14 * U_{perceptive} + 0.14 * U_{autonomous} + 0.14 * U_{collaborative} + 0.14 * U_{capability} + 0.14 * U_{misconception}$. For more information about the correction adaptation agents, the bidding process and the metrics, please see our previous paper[5].

3 System implementation

We start this section by describing the overall process in Kavanah and then describe in detail how the AUI helps the system build the adapted query. Kavanah consists of five modules as shown in Figure 1(a). The input module accepts the user's natural language queries and transfers them to the query module where they are parsed and converted into a *query graph* (QG) which is similar in construction to the user ontology network except that it may contain a node(s) representing a variable (usually denoted as X) that is necessary to represent unknown concepts in the user query. A query graph is a directed acyclic graph, where each node represents a concept or a relation among the concepts. A relation node should have concept nodes as parents and children. A concept node represents a noun phrase while a relation node represents a verb phrase in a user query or a natural language sentence. An example of a QG of the query "What causes liver damage?" is shown in the left side of Figure 3(b). The AUI uses the query graph and generates a new adapted query for the search module based on the current user model. An example of an adapted query is shown in the right side of Figure 3(b). The search module matches the QG of the adapted query against each *document graph* representing a record in the database of documents, chooses those records that have the number of matches greater than a user-defined threshold, and displays the output to the user. A document graph (DG) is a directed graph that contains concept and relation nodes and is also similar to the user ontology network (e.g Figure 2(a)). Note that all of the common concepts in all of documents are found in a global dictionary and domain ontology. A match between a QG and a DG is defined as the number of concept and relation nodes of the QG being found in the DG over the number of nodes of the QG. After the search module returns the search results, the feedback module allows the user to indicate whether the search result is relevant or not.

The AUI constructs the adapted query in Kavanah by maintaining the updated user interests, preferences and context. The logical architecture of the AUI is shown in Figure 1(b). The AUI determines the current interests by reasoning over the user ontology network with the concepts found in the user query set as evidence. Each element of the interest relevancy set consists of an *interest concept* and an *interest level*. The interest concept represents the topic that the user is currently interested in. It is determined from the user query and the set of documents that the user has indicated as relevant in the recent search. The interest level is a real number indicating how much the user is

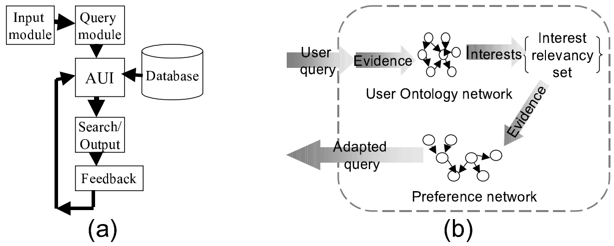

Figure 1. (a) The overall Kavanah architecture. (b) AUI Architecture

interested in the corresponding interest concept. Denote each interest concept as a and its associated interest level as $L(a)$. We compute $L(a)$ after every query by: $L(a) = 0.5(L(a) + \frac{n}{m})$ with n as the number of relevant documents containing this concept a and m as the number of relevant documents. If $L(a)$ falls below the user-defined threshold value, the corresponding interest concept a is removed from the interest relevancy set. To compute the new set of interests, we set as evidence in the user ontology network those concepts found in the query and the interest relevancy set, and perform belief updating on the user ontology network.

We construct the user ontology network dynamically by finding a common set of subgraphs of all relevant documents. Each document is represented as a DG(e.g Figure 2(a) and 2(c)). For each relevant document, we build a set of its subgraphs. A subgraph X of a DG Y is a DG such that each node a belongs to X also belongs to Y. The sets of subgraphs of the concepts "urate oxidase" and "cosmids" are generated in Figure 2(b) and 2(d). After generating all of the subgraphs, we compute the number of occurences for each subgraph in the entire set. We select those subgraphs that have the number of occurences greater than a user-defined threshold and consider them as the common subgraphs of the relevant documents. The common set of subgraphs of the two above concepts is shown in Figure 2(f). This set is used by an agent to update the user ontology network. The agent will check if a subgraph is not currently in the user ontology network, and adds it accordingly. This agent will ensure that the update will not result in a loop in the existing ontology network. If it does, we skip this addition. A new link between two existing concepts in the user ontology network will also be created if two concepts are linked in the set of common subgraphs and the frequency of these links exceeds a certain user-defined threshold.

The preference network is built when the user issues a new query and gives relevance feedback. Each user query is considered as a pre-condition

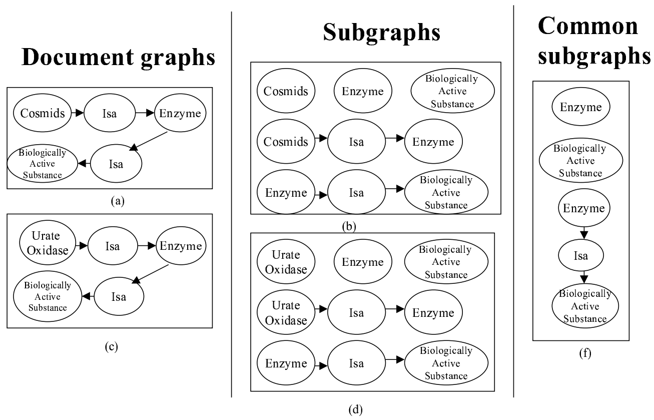

Figure 2. (a) The graph represents "cosmids". (b) Subgraphs of concept "cosmids". (c) The graph represents "urate oxidase". (d) Subgraphs of concept "urate oxidase". (f) The set of common subgraphs of the concepts "cosmids" and "urate oxidase".

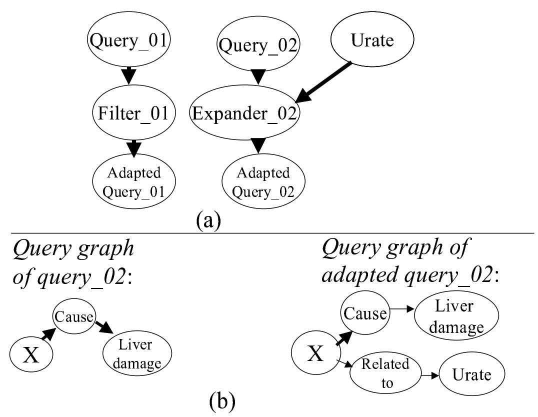

Figure 3. (a) An example of a preference network. (b) Examples of query graphs associated with the user query and the adapted query generated by AUI

node in the preference network. If this query or its part is already asked, the existing node in the preference network which has a QG matched with the QG of the new query or of its part will be set as evidence. Each interest concept from the interest relevancy set is added to the preference network as pre-condition node and set as evidence. If the user query is totally new, the tool being used by the user is set to the default value (a filter) and a goal node representing the filter tool is added to the preference network. Otherwise, it is set to the tool being represented by the goal node with highest utility value. Each action node represents a way to construct a adapted query based on the current tool, interests and user query. Figure 3(a) shows an example of a preference network in which the user is using an expander, is currently interested in the concept "urate" and wants to find out the causes of the liver damage. Note that each user query and adapted query generated by AUI associates with a QG. The current way of forming a new query is determined as a result of reasoning over the preference network by taking top n goals nodes as defined by the user. We computed the expected utility for a goal g with associated action a_i as follows: $EU(g) = P(g = T)U(\alpha - g = T) + P(g = F)U(\alpha - g = F)$ in which $U(\alpha - g = T/F) = \sum P(\alpha - g = T/F)U(a_i)$ and $U(a_i) = \sum U(attributes)$ The set of attributes in the formulas above includes the workload, user expertise, temporal memory, and spatial memory. For more detail information on this, please refer to our ealier work[6].

4 Empirical Evaluation

We empirically evaluate the system using the definitions of 100 concepts extracted from the Unified Medical Language System (UMLS). In the first evaluation, we focus on the quality of the retrieval process. We constructed a set of queries and processed this set through the system with and without the AUI. In this query set, we are mainly using the "wh" questions to find out the definitions of concepts or identify concepts that match certain requirements. For example, "what is urate oxidase?" or "which enzyme inhibits monoami oxidase and causes liver damage?". We made an assumption that the user does not just explore the concept randomly, but focuses on what he is studying. We used the precision and recall metrics commonly used in information retrieval[14] as our evaluation criteria. Figure 4 shows the precision and recall for all the questions in the cases with and without AUI. As we see, the precision and recall in cases that have an AUI are better than those without any help. If Kavanah is working without an AUI, it simply matches the QG of the user query with each DG representing each record in the database. Depending on how well the user manipulates the keywords in a query, the search may return more, less or even none of documents. This process requires the user either know the contents of the database or be very familiar with the search topics to achieve a decent result. The user's feedback is not used to adapt the search query. With AUI, depending on the user's feedback, Kavanah helps the user construct an appropriate search query that satisfies the user's searching intent. For example, if the user does not indicate any documents from

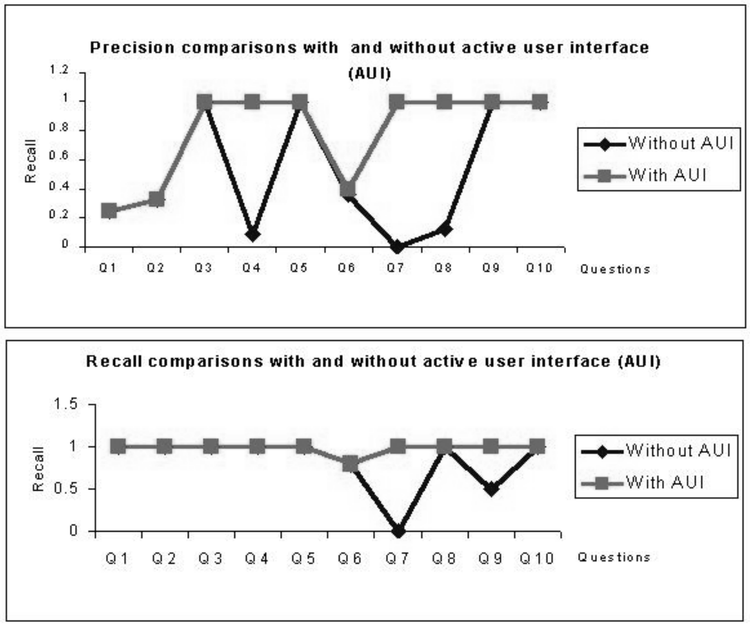

Figure 4. Precision and recall for Kavanah using with and without active user interface.

the returned list relevant, Kavanah then knows that perhaps, a wrong tool has been used, or the interests are not up-to-date or the ontology is far off the mark. It will automatically correct those misses in order to improve the quality of the search.

We also evaluated the process of constructing and updating the user's ontology network by building simulated user ontologies from the domain ontology. We randomly choose some concept nodes from the domain ontology (referred in this experiment as *testing concepts*) and randomly remove some links associated with them to see if our system can reconstruct those missing links in the user ontology network. For each testing concept, we construct a set of queries such that they reflect the relations between the testing concept and the removed links. We compute the link error as follows: $LinkError = \frac{n}{m}$ in which n is the number of links in the user ontology network matched against the target user subgraph's and m is the total number of links of the user ontology network constructed by AUI. First, we performed this experiment using the testing database mentioned above and found out that there is a large mismatch between the domain ontology and the set of concepts being

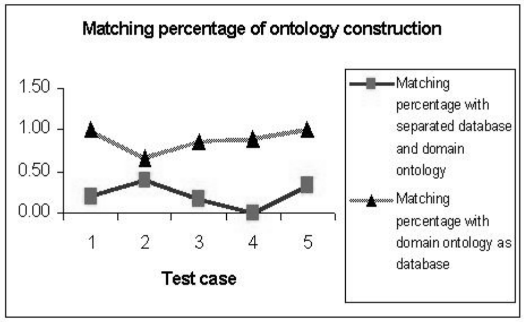

Figure 5. Link matching percentage of ontology construction

used to construct our testing database. As it is shown in Figure 5, the matching percentage of the links created in the user ontology against the real user subgraph in domain ontology is low for this case. Next, we tried to use a portion of the domain ontology as the testing database to perform the same experiment in order to see if the matching percentage of those links is changed without the mismatch between the database and domain ontology. Obviously, we see that the results are significantly better.

5 Related work

In the information retrieval and information filtering community, dynamically reacting to the changes in the user interests, preferences and context (i.e., ontology) to reduce the user cognitive workload is a challenging problem. Most of the research from information retrieval has been focused on capturing the user interests only[7,18,3]. Work by the InfoSleuth Group explores the ontology to improve the retrieval process[10] is a closely related to our work. In this work, ontology is dynamically constructed from the text and constantly been updated as the retrieval process goes on. However, this technique doesn't focus on the preferences and interests. There is not enough detail from the paper about this approach to empirically compare it with ours. Another work related to ours is IRIA[1] which unobtrusively builds a map of accessible relevant information and uses it to enable users to find information quickly. The difference between this work and ours was the IRIA approach is based on a context-sensitive search which spreads the user interests based on the relevant information to users while ours is based on a decision theoretic approach to maintain the model of the user preferences, interests and context.

6 Future work

This paper has described our on-going work to construct an active user interface that provides intelligent assistance to the user in an information retrieval system. There are a number of issues that arise from our design and empirical evaluation. We want to extend our evaluation to a more complex scenario with different kind of questions and search strategies. Unfortunately, the current database has the problem of low term frequency which is usually referred to as data *sparseness problem* in information retrieval[17]. We are also looking for another supplement database or semantic network in UMLS that will help us to overcome the problem of disjointness between the domain ontology and the database used as testbed. We wanted to measure not only links errors, but also concepts errors which refers to the number of concepts in the user ontology network matched against the original real user subgraph. At present, we use a fading mechanism to fade away interests, preferences or context that are no longer used. This may result in more frequent updates than necessary if the user intent is not very dynamic. We wanted to employ a mechanism to differentiate between the short-term and long-term interests, preferences and context in an intuitive way using findings from experimental psychology[8].

Acknowledgements

This work was supported in part by AFOSR Grant No.F49620-00-1-0244 and the AFRL Human Effectiveness Directorate Through Sytronics Incorporated. Thanks to Greg Johnson and Thuong Doan for helping with this paper.

References

1. Anthony, Jr. G. F.; Devaney, M.; and Ram, A. 2000. Iria: The information retrieval intelligent assistant. In *Proceedings of the International Conference on Artificial Intelligence.* 275-280.
2. Balabanovic, M.; and Shoham, Y. 1997. Content-based, collaborative recommendation. In *Communications of the ACM*, 66–72. Vol 40, No.3
3. Billsus, D.; and Pazzani, M. J. 2000. User modeling for adaptive news access. In *Journal of User Modeling and User-Adapted Interaction.*Vol(10),issue 2/3.147-180.
4. Brown, S. M.; Santos, Jr. E.; and Banks, S. B. 1999. Active user interfaces for building decision-theoretic systems. In *Proceedings of the 1st Asia-Pacific Conference on Intelligent Agent Technology, Hong Kong.* 244-253.
5. Brown, S.M.; Santos, Jr. E.; Banks, S. B.; and Oxley, M. 1998. Using

explicit requirements and metrics for interface agent user model construction. In *Proceedings of the Second International Conference on Autonomous Agents*, 1-7, Minneapolis, MN.
6. Brown, S. M. 1998. Decision theoretic approach for interface agent development. Ph. d disertation.
7. Chen, L.; and Sycara, K. 1998. Webmate: A personal agent for browsing and searching. In *Proceedings of the 2nd International Conference on Autonomous Agents and Multi Agent Systems.* Minneapolis, MN.
8. Ericsson, K. A.; and Kintsch, W. 1995. Long-term working memory. In *Psychology Review*, 211–245. 102.
9. Horvitz, E.; Breeze, J.; Heckerman, D.; Hovel, D.; and Rommelse, K. 1998. The lumiere project: Bayesian user modeling for inferring goals and needs of software users. In *Proceedings of the Fourteenth Annual Conference on Uncertainty in Artificial Intelligence.*256-265.
10. Hwang, C. H. 1999. Incompletely and imprecisely speaking: Using dynamic ontologies for representing and retrieving information. InfoSleuth Group, Microeletronics and Computer Technology Corp, 3500 West Balconnes Center Drive, Austin, TX 78759.
11. Maes, P. 1994. Agents that reduce work and information overload. In *Communications of the ACM*, 31–40. 37(7).
12. Geddes, N. 1986. The Use of Individual Differences in Inferring Human Operator Intentions. In *Proceedings of the Second Annual Aerospace Applications of Artificial Intelligence Conference*
13. Gruber, T. R. 1993. Toward principles for the design of ontologies used for knowledge sharing. In *the International Workshop on Ontology.*
14. Salton, G.; and McGill, M. 1983. *Introduction to Modern Information Retrieval.* McGraw-Hill Book Company.
15. Santos, Jr. E.; Brown, S. M.; Lejter, M.; and Banks, S. B. 1999. Dynamic user model construction with bayesian networks for intelligent in formation queries. In *Proceedings of of the 12th Flairs conference.*3-7.
16. Santos, Jr. E.; Nguyen, H.; and Brown, S. M. 2000. Medical document information retrieval through active user interfaces. In *Proceedings of the 2000 International Conference in Artificial Intelligence.*323-329.
17. van Rijsbergen, C. 1975. *Information Retrieval.*The Whitefriars Press Ltd, London and Tonbridge.
18. Widyantoro, D. H.; Ioerger, T. R.; and Yen, J. 1999. Adaptive agent for learning changes in user interests. In *Proceedings of the International Conference on Information and Knowledge Management CIKM'99.* Kansas City.