Intelligent Systems Program Comprehensive Examination

Intelligent Interface

Noboru Matsuda
March 14, 2003

Question from Peter Brusilovsky:

Write a short essay (no more than 8 pages) that answers both questions below:

#1. Adaptive Hypermedia systems are to guide their users to the information that is most suitable to their knowledge, goals and interests. At the same time, these systems can only work in the closed collections of documents that were manually indexed to connect documents and knowledge. From another side, adaptive information filtering and retrieval systems are also able to recommend relevant documents that were never indexed manually. Does it mean that in guiding the user to relevant information adaptive information filtering and retrieval is better to adaptive hypermedia in all aspects? Are there aspects in which adaptive hypermedia is superior to information filtering and retrieval? Review a number of known techniques in different areas of intelligent interfaces and discuss how these techniques can be used to empower existing adaptive information filtering and retrieval systems so that they can match some unique powers of adaptive hypermedia.

#2. An improvement of on-line help systems has been a long term concern of researchers in the field of intelligent interfaces. This problem has been addresses by different groups of researchers working on different aspects of intelligent interfaces. Provide a brief review of the work on "better online help" known to you from the studied literature. Discuss relative merits of the known approaches. Now try to use your knowledge of several techniques from different subareas of the field of intelligent interfaces to draft the imaginary "intelligent help" that is based on at least three different ideas comig from different research streams in intelligent interfaces.

Note:

To answer the given questions, I read additional three papers that are not in the reading lists: (Brusilovsky, 1996; Encarnacao & Stoev, 1999; Jerrams-Smith, 2000)

Answer for question #1:

The question apparently expect me to compare *adaptive hypermedia systems* that are supposed to work on a closed corpus and *adaptive information filtering-&-retrieval systems* that are supposed to work on an open corpus, and discuss advantages and disadvantages over these two types of adaptive systems.

This distinction sounds confusing to me, because information filtering-&-retrieval can be done on a close corpus as well (e.g., information filtering for listserver, which is closed with respect to the source of the documents), and the documents scattered over the Web, *the* open hyperspace, can be weaved into an adaptive hypermedia tailored to individual users with the links and contents gathered by the technique used for information filtering and retrieval (i.e., an adaptive hypermedia system can be built for an open corpus with the technique used for information filtering and retrieval). Furthermore, for some scholars, information filtering system is inevitably adaptive; information filtering system necessarily has a user profile hence, by definition, must be adaptive (Hanani, Shapira, & Shoval, 2001, p.204). Hence, discussing the "adaptiveness" of information filtering-&-retrieval systems also sounds confusing to me.

Above observation makes me very cautious not to compare apples and oranges. Anyhow, I started to define "an adaptive hypermedia system" as an adaptive system that works in a closed hyperspace where manual indexing is possible, whereas "an adaptive information filtering and retrieval system" as an adaptive system (that does something that may or may not equivalent to the ones done by adaptive hypermedia systems) that works in an open hyperspace where manual indexing is not conceivable. The motivation of these definitions was that such an interpretation best fits in with the exam question.

I then had a great difficulty to understand different between "open" and "closed" corpora. In other words, I did not understand what makes "open" corpus open. Even when an adaptive hypermedia system works on a fixed number of hypermedia documents (hence they are closed with respect to their *location*), the contents of the documents might be modified hence they are open with respect to their *content*, just like the target corpora of email-filtering systems (i.e., incoming emails) are commonly believed to be open due to its unexpectedness in the contents of email messages. Nonetheless, nobody would be happy to claim that an adaptive hypermedia system working on a local PC (again, close to its location and open to its contents) is taking care of an "open" hyperspace.

Is that just a matter of quantity of information to be processed? No. Google, the world leading search engine, works on the whole Web space, which is undoubtedly "open." Even though working on an open corpus, Google indeed maintains the indexes and links for whole document on the Web by crawling over the world hence at least for Google the whole Web is closed at a time a query is posed by a user. So, the amount of information alone can not determine whether a corpus is open or close. Is it an essential characteristic of a given corpus that opens or closes the corpus? No. We can "close" an open corpus by restricting the search space to the hyperspace that is hang off the current browsing point (Brusilovsky, 2001, p.93).

This confusion motivated me to further elaborate the definition of adaptive hypermedia systems and adaptive information filtering-&-retrieval systems such that "an adaptive hypermedia system" is an adaptive system that provides adaptive hypermedia documents based upon a set of indexed documents, whereas "an adaptive information filtering and retrieval system" as an adaptive system that provides adaptive hypermedia documents without indexing.

Given above interpretation for the question #1, I will provide a brief overview for the adaptive hypermedia systems and adaptive information filtering-&-retrieval systems. I then provide a comparison between these two systems as an answer to the question #1.

1. Adaptive Hypermedia Systems

Adaptive hypermedia systems have been emerged as an expansion of "static" hypermedia systems, which were commonly built in a single (and isolated) PC. The major drawback of such a static hypermedia is an inability to change its performance adaptively to the users hence provide "all things to all people" (Brusilovsky, 2001, p.87). So, the primary motivation for adaptive hypermedia systems is to provide "needed things to individual people."

The range of task to which adaptive hypermedia systems can fit is quite broad. A recent review includes following systems (Brusilovsky, 2001): *Educational hypermedia systems* provide hypermedia documents that agree with individual student's competence and needs. *On-line information systems* select and present information (among many of them) that satisfies the information-seekers' (i.e., the users') need. *Information retrieval hypermedia systems* search document in a hyperspace, select only appropriate ones, structure them with appropriate annotations (if any), and display them to the users. *On-line help systems* watches the uses' use of application software, and provides appropriate help when needed.

Adaptive hypermedia systems take account of characteristics of *users* and *environments* to generate hypermedia documents adapted to individual uses. Characteristics of users are expressed by the user's intention (including the goal and the task), background knowledge (including their experience), preference, interests, and traits (i.e., personality factors, cognitive factors, learning styles, etc.). For example, characteristics of environments on which the web-based user applications run include (a) the location of users using applications, and (b) the platform, namely, the hardware specifications. MANNA, a map annotation assistant (Eisenstein, Vanderdonckt, & Puerta, 2001), helps users to make an annotated map that requires a user (a geologist) a variety of activities from going outside with a palm computer to writing a report with the annotated map with a desktop PC. Throughout the various activities, the user can work on single data (i.e., a map) but with the different platforms. MANNA automatically changes its appearance and performance to fit each platform's characteristics. For example, clicking on spot of the map allows a user to enter a new note. When working with a mobile PC with a small screen (e.g., PDA), no further activities appears. On the other hand, when working with a desktop PC, it also brings up a rich set of geographical and meteorological information describing the selected area. Although this is not an example of adaptive hypermedia system, but the same idea would apply for hypermedia documents.

Given the profile of a user and an environment, adaptive hypermedia systems produce hypermedia documents that are adapted to the user in the documents' appearance and structure. The former is called *content level adaptation* or *adaptive presentation*, which is commonly supported by two methods; text adaptation and multimedia adaptation. The latter is called *link level adaptation* or *adaptive navigation support*, which is commonly supported by five methods; link hiding, sorting, annotation, direct guidance, and hypertext map adaptation (Brusilovsky, 2001, p.97). Namely, adaptive hypermedia documents have only relevant links with relevant priorities. The links might be adaptively enhanced by adding annotated texts. The hyperspace itself might be adaptively organized for individual users.

The fundamental issue for the adaptive navigation support is to calculate "relevance" of the links, which represent the relevance of the hypermedia documents associated to the links. The major techniques to calculate the relevance of links are (1) numerical matrix, (2) rule-based assessment, and (3) conceptual network (Brusilovsky, 1996). As mentioned in the question, these techniques require all the links to be known at the time of calculation of relevance. Hence indexing is, also by definition, the central issue for adaptive hypermedia systems.

2. Adaptive Information Filtering and Retrieval

Information filtering let users to expose only to relevant information. Some examples of information filtering systems are filtering for search results, personal information filters based on personal profiles, listservers or newsgroup filters for groups and individuals, browser filters to block non-valuable and irrelevant information/contents, filters for e-commerce applications that address products and promotions to potential customers only (Hanani et al., 2001). The common challenge among those tasks lies in its tremendous amount of information to explore and continuous emergence of new information, which shut off indexing.

Quite interestingly, according to a recent review on information filtering (Hanani et al., 2001), the same techniques used for adaptive hypermedia systems have been used for information filtering, namely, (1) *statistical matrix*, which calculate correlations (or Cosine measure) between the user's profile and the document's vector, (2) *AI approach*, which utilizes rule-base, semantic-net, or neural networks, and evolutionary genetic algorithm.

With above techniques, adaptive information filtering and retrieval systems analyze incoming information either (1) to measure a "similarity" (or, "distance," if you will) of the information with known documents, (2) to determine a relevance of information in terms of predefined criteria, and (3) to categorize it into predefined information categories.

There are two basic types of adaptive information filtering and retrieval systems (Brusilovsky, 2001): (1) *Search oriented systems* create a list of links to documents that satisfy the user's current information request. In search oriented systems, filtering do more than "one-shot" Web-search by applying *link removal* and *link annotation*. (2) *Browsing oriented systems* integrate retrieved information to best fit user's activities. Apparently, some methods for adaptive navigation support mentioned in the previous section discussing adaptive hypermedia systems apply for adaptive information filtering and retrieval systems with the support of techniques listed above. We discuss this point in the next section.

3. Comparison between Two Adaptive Systems

One of the potential strength of adaptive hypermedia systems is that they can proactively provide appropriate information to the uses. This can be done because adaptive hypermedia systems maintain indexes of the whole hypermedia document in hand. They can monitor user's activities to build a user mode, and can intervene in the user's task with the *most* useful information. On the other hand, adaptive information filtering and retrieval systems can provide information only upon users' requests and the quality of information is restricted in a sense that they are only guaranteed to satisfy certain criteria; they never know if there exists better information somewhere in the world.

One way to improve the performance of information filtering and retrieval systems is to use *stereotypic profiles* as the default profiles for incoming (emerging) information (Hanani et al., 2001). An example of this technique is a filtering system with two filtering phases; remote phase and local phase (Kay and Kummerfeld, 1995 cited in Hanani et al., 2001). Remote phase builds stereotypes of users from a very large database items. Once the stereotypes are built, then information necessary for individual users are matched against the items in stereotypes at the local phase of filtering.

Similar technique can be used to read a novel hypermedia document, analyze it, and relate it with a set of known documents. A potential application of this idea is to put an annotation for educational hypermedia system. With a pool of "local" documents such system can analyze any documents by the students (i.e., instructional materials) in the Web searched, and annotate notes on the searched document to the locally pooled documents.[1]

Answer for question #2:

On-line help systems apparently have "received almost no attention from adaptive hypermedia researchers in the last few years. Apart from an extended version of ORIMUS (Encarnacao & Stoev, 1999) we can cite no new interesting systems (Brusilovsky, 2001, p.90)." Nonetheless, I started to read two on-line help related papers (Encarnacao & Stoev, 1999; Jerrams-Smith, 2000), in which I learned a non-trivial relationship between on-line help systems and adaptive learning support systems (i.e., intelligent tutoring systems). Hence, in the brief review on the on-line help systems, I include a study conducted by Lester et al on the lifelike pedagogical agent (Lester, Stone, & Stelling, 1999).

1. Brief Review on Intelligent Help

An on-line help system, to my best knowledge from the readings, places between a user and application software, monitors the user's activities, passes the user's actions to application software if it does not

[1] Actually such system has been developed and published recently in Japan. It is truly unfortunate that I can not cite their work here merely because it is not written in English.

brings fatal catastrophe (e.g., deleting an important file), and provides helps (e.g., warning, correction, and instruction, etc.) when the user exhibits an inappropriate action. The helps provided to the users might be adjusted to the users' skill level as well as to the task context. The helps might not be only to prevent / recover from an error, but also to lead the users to their goals. The helps can be both directive and non-directive. To provide these various kinds of helps, the on-line help systems commonly maintain a model of users as well as a model of the target task.

An UNIX on-line help system developed by Jerrams-Smith (2000) is aimed to assist novice users in working with a UNIX operating system. The system has a knowledge base describing rules to provide helps. The system also has an elaborated error detecting mechanism that can discriminate mistypes and misunderstanding.[2]

ORIMUHS (Encarnacao & Stoev, 1999) is a generic on-line help system aimed work with GUI-based application software. ORIMUHS monitors user's every "action," which is a recordable event in the GUI caused by the user (e.g., mouse movement, selection, keyboard input, etc.). The task is modeled hierarchically to represent relationship between the goals (and/or intentions) of the user and the operations to achieve a goal. The competence (or, skill level, if you will) of the user is represented with two different user models; *global use information* that represents overall competence maintained across the different tasks, and *working-context specific user information* that represents competence of each skills assessed against a specific task. The similar technique is adopted in PACT tutors, namely, the knowledge trace (global), which represents use's competence level aggregate in the past experience on the different problems, and the model trace (local), which represents use's competence on the current problem.

The lifelike agent (Lester et al., 1999) is designed to assist students learning by problem solving. The agent is implemented as an animated character that can move around a learning environment and maintain a mixed-initiative dialogue. The task for students to learn is rather simple[3], but the agent has a set of

---

[2] Very little is explained about this technique in the paper. But the technique to analyze use input seems to be supported by empirical data on novice user's performance.

[3] Basically, the students must design a plant that is robust enough to grow in a given environment (out of 16 different environments). A plant is designed with different root, stem, and leaves. There are 8 types of root, 8 types of stems, and 8 types of leaves.

elaborated strategies to control the initiative transfer between a student and the agent. Although the agent is not designed to provide on-line help for application software, it has the same characteristics as intelligent on-line help systems; act as a mediator, maintains a user (i.e., a student in this case) model, has a set of strategies to provide appropriate helps.

In sum, the common features in those three on-line help systems can be reviewed as follows. They monitor the user's performance over the shoulder, and intervene in the user's activity when necessary – *proactive helper*. They can also provide helps as a response of the user's question – *passive helper*. They educate the users not to repeat the same error – *peer tutor*. They maintain a user model to adapt their performance – *assessor*. They hold expertise (i.e., a domain model) in the target task domain – *domain expert*.

With my full imagination and knowledge learned from the readings, I will now expand these distinctive features to design a better, indeed ideally, on-line help agent.

2. Gatta be Smart, KlipGuy!

In the following imaginary story, known techniques commonly used in intelligent user interface are *italicized*.

The imaginary task domain is ordinal applications working commonly on personal computers, namely, a word processor, a spread sheet, a database management system, a presentation tool, a drawing tool, and things like that. The imaginary intelligent help agent is supposed to assist users to work with these "complicated" applications. Let's call this agent KlipGuy (rhymes with /clip guy/, who is working around in a famous GUI environment, and they could be actually very similar in their appearance, but KlipGuy should be much smarter in a sense described below).

Just like the lifelike agent (Lester et al., 1999), KlipGuy must be implemented as an *animated agent* that can communicate with the uses but in *spoken dialogue* while *adaptively control its initiative*. The adaptive spoken dialogue system is still developing, but there are some known practical techniques such as using a regression model to determine user satisfaction, which in turn determines initiative shift (Litman & Pan, 2002).[4]

KlipGuy not only provides on-line help based on a user and a domain model (the common feature in on-line help system discusses in the previous section), but also refer to external information on the Web. Specifically, KlipGuy might suggest the user to specify, as a set of key words, a concept that the user cannot understand or a goal that the user has difficulty to achieve. Those keywords are uses to *search relevant document* within, say, FAQ database. *Information-filtering technique* might further elaborate the quality of information provided to the user. Results must be displayed as an *adaptive hypermedia document* with only links and annotations relevant to the user.

KlipGuy would also encourage the user to contribute to the FAQ database by posting the user's experience (i.e., an episode of problem solving) as *shared information*. KlipGuy must maintain the FAQ database in a way that closely related information must placed together under an appropriate structure, just like the Nugget server does (Goecks & Cosley, 2002). Using the same technique utilized in NuggetMine, the information stored in the FAQ database can be *distributed to the users with the same interests*. *User profiling* then would be the key issue for such proactive information sharing.

What about automatic generation of *multimedia document*? It would facilitate users to understand the help if KlipGuy automatically generates multimedia documents depending on the *user preference*. Some users might prefer to read just texts, some might have tendency to like more audio-visual oriented documents. Although Maybury (2001) mentioned a concept of *multimodal output generation*, I could not find any robust technique for this issue in the readings.

---

[4] Litman et al found that the user satisfaction (US) is a function of misrecognition, turns, rejections, and timeouts: US = -0.67 * N(Misrecognition) -0.39 * N(Turns) -0.33 * N(Rejections) -0.19 * N(Timeout), where N() is a normalization function.

Reference:

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction, 6*(2-3), 87-129.

Brusilovsky, P. (2001). Adaptive hypermedia. *User Modeling and User Adapted Interaction, 11*(1-2), 87-110.

Eisenstein, J., Vanderdonckt, J., & Puerta, A. (2001). Applying model-based techniques to the development of UIs for mobile computers, *Proceedings of the 6th international conference on Intelligent user interfaces* (pp. 69-76). New York, NY: ACM Press.

Encarnacao, L. M., & Stoev, S. (1999). An Application-Independent Intelligent User Support System Exploiting Action-Sequence Based User Modelling. *User Modeling: Proceedings of the Seventh International Conference*, 245-254.

Goecks, J., & Cosley, D. (2002). NuggetMine: intelligent groupware for opportunistically sharing information nuggets, *Proceedings of the 7th international conference on Intelligent user interfaces* (pp. 87-94). New York, NY: ACM Press.

Hanani, U., Shapira, B., & Shoval, P. (2001). Information Filtering: Overview of Issues, Research and Systems. *User Modeling and User-Adapted Interaction, 11*(3), 203-259.

Jerrams-Smith, J. (2000). An intelligent human-computer interface for provision of on-line help. *Artificial Intelligence Review, 14*(1-2), 5-22.

Lester, J. C., Stone, B. A., & Stelling, G. D. (1999). Lifelike pedagogical agents for mixed-initiative problem solving in constructivist learning environments. *User Modeling and User Adapted Interaction, 9*(1-2), 1-44.

Litman, D. J., & Pan, S. (2002). Designing and Evaluating an Adaptive Spoken Dialogue System. *User Modeling and User-Adapted Interaction, 12*(2-3), 111-137.

Maybury, M. T. (2001). Intelligent User Interfaces for All. In C. Stephanidis (Ed.), *User Interfaces for All: Concepts, Methods, and Tools* (pp. 65-80). Mahwah, NJ: Erlbaum.

Appendix: I: Assigned Reading List

General Principles

Olson, G. M., & Olson, J. S. (2003). Human-computer interaction: Psychological Aspects of the Human Use of Computing. Annual Review of Psychology, 54, 491-516.

Maybury, M. T. (2001). Intelligent User Interfaces for All. In C. Stephanidis (Ed.), User Interfaces for All: Concepts, Methods, and Tools (pp. 65-80). Mahwah, NJ: Erlbaum.

Roth, E. M., Malin, J. T., & Schreckenghost, D. L. (1997). Paradigms for Intelligent Interface Design. In M. G. Halander & T. K. Landauer & P. V. Prabhu (Eds.), Handbook of Human-Computer Interaction (2nd ed., pp. 1177-1201). Amsterdam: Elsevier.

Dahlback, N., Jonsson, A., & Ahrenberg, L. (1993). Wizard of Oz Studies - Why and How, Proc. of the International Workshop on Intelligent User Interfaces (pp. 193-199), Orland, FL: ACM.

Adaptive Systems

Brusilovsky, P. (2001). Adaptive hypermedia. User Modeling and User Adapted Interaction, 11(1-2), 87-110.

Goecks, J., & Shavlik, J. (2000). Learning Users' Interests by Unobtrusively Observing Their Normal Behavior, Proceedings of the 5th international conference on Intelligent user interfaces (pp. 129-132). New York, NY: ACM Press.

Jameson, A., Schafer, R., Weis, T., Berthold, A., & Weyrath, T. (1999). Making Systems Sensitive to the User's Time and Working Memory Constraints, Proceedings of the 4th international conference on Intelligent user interfaces (pp. 79-86). New York, NY: ACM Press.

Navigation / Information Filtering / Information Retrieval

Padovani, S., & Lansdale, M. (2003). Balancing search and retrieval in hypertext: context-specific trade-offs in navigational tool use. International Journal of Human-Computer Studies, 58(1), 125-149.

Budzik, J., Bradshaw, S., Fu, X., & Hammond, K. J. (2002). Supporting on-line resource discovery in the context of ongoing tasks with proactive software assistants. International Journal of Human-Computer Studies, 56(1), 47-74.

Hanani, U., Shapira, B., & Shoval, P. (2001). Information Filtering: Overview of Issues, Research and Systems. User Modeling and User-Adapted Interaction, 11(3), 203-259.

Collaboration and Sharing

Goecks, J., & Cosley, D. (2002). NuggetMine: intelligent groupware for opportunistically sharing information nuggets, Proceedings of the 7th international conference on Intelligent user interfaces (pp. 87-94). New York, NY: ACM Press.

Lieberman, H., Van Dyke, N. W., & Vivacqua, A. S. (1999). Let's browse: a collaborative Web browsing agent, Proceedings of the 4th international conference on Intelligent user interfaces (pp. 65-68). New York, NY: ACM Press.

Communicative Agents / Natural Language Interfaces

Litman, D. J., & Pan, S. (2002). Designing and Evaluating an Adaptive Spoken Dialogue System. User Modeling and User-Adapted Interaction, 12(2-3), 111-137.

Souvignier, B., Kellner, A., Rueber, B., Schramm, H., & Seide, F. (2000). The thoughtful elephant: strategies for spoken dialog systems. IEEE Transactions on Speech and Audio Processing, 8(1), 51-62.

Lester, J. C., Stone, B. A., & Stelling, G. D. (1999). Lifelike pedagogical agents for mixed-initiative problem solving in constructivist learning environments. User Modeling and User Adapted Interaction, 9(1-2), 1-44.

Model-based Interface

Eisenstein, J., Vanderdonckt, J., & Puerta, A. (2001). Applying model-based techniques to the development of UIs for mobile computers, Proceedings of the 6th international conference on Intelligent user interfaces (pp. 69-76). New York, NY: ACM Press.

Derthick, M., & Roth, S. F. (2001). Example based generation of custom data analysis appliances, Proceedings of the 6th international conference on Intelligent user interfaces (pp. 57-64). New York, NY: ACM Press.

Puerta, A., & Eisenstein, J. (1999). Towards a general computational framework for model-based interface development systems, Proceedings of the 4th international conference on Intelligent user interfaces (pp. 171-178). New York, NY: ACM Press.

Interface that Learns

Wolfman, S. A., Lau, T., Domingos, P., & Weld, D. S. (2001). Mixed initiative interfaces for learning tasks: SMARTedit talks back, Proceedings of the 6th international conference on Intelligent user interfaces (pp. 167-174). New York, NY: ACM Press.

McDaniel, R., & Myers, B. A. (1998). Building Applications Using Only Demonstration, Proceedings of the 3rd international conference on Intelligent user interfaces (pp. 109-116). New York, NY: ACM Press.

Visualization

Jung, T., Gross, M. D., & Do, E. Y.-L. (2002). Annotating and Sketching on 3D Web Models, Proceedings of the 7th international conference on Intelligent user interfaces. New York, NY: ACM Press.

Hsieh, H.-W., & Shipman, F. M. (2000). VITE: A Visual Interface Supporting the Direct Manipulation of Structured Data Using Two-Way Mappings, Proceedings of the 5th international conference on Intelligent user interfaces. New York, NY: ACM Press.

Kerpedjiev, S., & Roth, S. F. (2000). Mapping Communicative Goals into Conceptual Tasks to Generate Graphics in Discourse, Proceedings of the 5th international conference on Intelligent user interfaces. New York, NY: ACM Press.

Multimodal Interface

Tanaka, K. (1999). A robust selection system using real-time multi-modal user-agent interactions, Proceedings of the 4th international conference on Intelligent user interfaces (pp. 105-108). New York, NY: ACM Press.

Appendix II: Supplement papers read after received the questions

Adaptive Systems

Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction, 6(2-3), 87-129.

Intelligent Help

Jerrams-Smith, J. (2000). An intelligent human-computer interface for provision of on-line help. Artificial Intelligence Review, 14(1-2), 5-22.

Encarnacao, L. M., & Stoev, S. (1999). An Application-Independent Intelligent User Support System Exploiting Action-Sequence Based User Modelling. User Modeling: Proceedings of the Seventh International Conference, 245-254.