The Importance of Contexts in Mixed-Initiative Interactions for Online Help

Shilpi Rao[#], Samir Menon[$], Jurika Shakya[#], Mayo Jordanov[#], Liam Doherty[#], Vive Kumar[#]
#School of Interactive Arts and Technology, Simon Fraser University, Canada
$Indian Institute of Information Technology, Allahabad, India

srao@sfu.ca, vive@sfu.ca

Abstract

Online help technologies range from sophisticated graphical interfaces that guide users, to proactive and intelligent tutorial interactions. Introducing ready, able, and willing human helpers in help scenarios has proven to be an important milestone in help technologies. In this paper we argue how techniques of mixed-initiative interactions can be successfully deployed in online help. We contend that a well-defined context, that encapsulates the relative knowledge, preferences, and task goals of the helper and the helpee, is integral to the success of mixed-initiative help interactions. We present empirical results to highlight the need for context-awareness in help scenarios and argue how such contexts dynamically regulate the contributions of the conversants, the helper, the helpee, and the help system, in mixed-initiative interactions.

Research Background

Online-Help technologies range from sophisticated graphical interfaces that guide users, to proactive and intelligent tutorial interactions. They could be passive or active, provided with canned solutions or knowledge-based inferences, generic or task specific, collaborative or autonomous, centric or distributed. Introducing ready, able, and willing human helpers in the help scenario has proven to be an important milestone in help technologies and it sets the stage up for the introduction of mixed-initiative interactions in such scenarios. We present an overview of contemporary help systems in the next two sections.

Mixed-initiative interactions attempt to model a middle-ground interaction strategy between AI and HCI where conversants (agents and users) contribute appropriate information when it is best suited and towards mutually negotiated goals (Hearst 1999). Also, depending on the needs of the helpee, the roles of a helper or a help system can be opportunistically negotiated (Allen 1999). These negotiations, based on a common contextual understanding, can determine which conversant has the control of the help conversation, at what point in time, and on what basis. A common contextual understanding should encapsulate the relative knowledge, preferences, and task goals of all the conversants in a help scenario. We present a functional description of a help context and outline some of the fundamental characteristics of mixed-initiative systems in later in the paper.

One of the key characteristics of mixed-initiative systems focuses on the explicit representations for initiatives. Most help systems reported in the literature of Intelligent Tutoring Systems neither explicitly represent opportunities for initiatives nor regulate the interactions based on which conversant has the initiative. The second section of this paper outlines a simple framework that explicitly represents contextual information and utilizes the same in promoting initiative-taking among the conversants. The framework has been implemented in a system named Helper's Assistant and has been empirically tested to determine its utility in a real-world scenario. The design of the empirical study and some of the key results of the study have been reported under the third section. In conclusion, we highlight the impact of formalized contextual information in mixed-initiative help scenarios.

Help Systems

Help systems have been extensively investigated for some time now. Houghton (Houghton 1984) reports different types of early help systems that include "command and help assistance", "error prompting", "online tutoring", "online documentation", and "help scripting". Online help investigations also include the impact of display format (Cherry et al. 1989, Gwei 1990), animation (Coffee 1997), graphics (Wise 1993), and hypermedia (Lee 1998) on help systems. Most contemporary software tools have generic help facilities including metaphoric help (user-friendly interfaces) and online help (www manuals). A few of them have context-specific help, as in Lumiere Project (Horvitz et al. 1998). These approaches, in an attempt to be self-sufficient, do not consider human-help as a resource at all.

Human help is inherently personalized, customized, and delivered exactly when needed. Help technology being investigated under the aegis of Intelligent Tutoring Systems is insufficient to duplicate the sophistication and depth of human help (Kumar 2004, Kumar et al. 2004). Such attempts are limited by the shortcomings of the context information, the inability to match a help request to an appropriate help response, and the inadequacy in meeting time limitations. An ideal system would attempt to store or generate vast amounts of situated and individualized help information and to provide fast and structured access to it. Yet, such an approach introduces unmanageable computational complexity, inadequate failure handling, deficient self-improvement, and inflexible generalization.

These shortcomings have been addressed to a greater extent in recent help systems where human helpers are introduced as an integral part of the help system (Collins et al. 1997, Greer et al. 1998), aptly named Human-In-The-Loop approach. A typical help scenario involves a user and the help system, where help is delivered thorough a dialog between the user and the help system. The human-in-the-loop approach brings in human helpers to complement the system. A helpee consumes help and a helper provides help. Adding a human helper to the mix can buttress the help system when it fails. This can be particularly beneficial if the conversations among the helper, the helpee, and the help system are dynamically regulated based on the principles of mixed-initiative interactions.

Human-in-the-loop

Human help is superior to machine help as long as the helper is competent and pertinent context is established between the person delivering help and the person receiving help. This is true because human helpers understand subtle contextual cues better than any help system and identify and deliver help responses within a reasonable time limit. Successful peer help among friends and colleagues is due to the establishment of personal context. A context is a shared understanding of the help requirement. Establishing a suitable context is the heart of the problem in machine help.

Kumar (Kumar 2001) explores techniques and interfaces to support the human helper who has been embedded in a human-computer help environment, where the design of the help system is capable of acquiring context information, making useful knowledge-based help responses, and ensuring delivery of help within acceptable time limits. Human-in-the-loop approach, aided by task-specific user-centric contexts, can assist the development of a pragmatic help system that is intelligent, informed of the user, informed of the tasks involved, informed of the information used, informed of the collaborative interactions, and informed of the help resources.

Contexts

Most contemporary help systems are content-rich and context-poor. That is, a help request can be resolved using a variety of information and tools, hence content-rich, but it is a difficult problem to deliver the help in a personalized fashion target to the user's needs, hence context-poor.

A context gives depth to information using which a help context facilitates the helper and the helpee to interact in a congenial fashion. Information contained within a context is, in most cases, localized. That is, the context information from one help session may not be relevant in another help session. Context information can be used during a help session in a variety of ways: to verify the suitability of the helper-helpee pair; to find out how much time the helper would like to help in a help session; to suggest help tools that the helpee would like to use; to suggest pedagogy (as part of the delivery of instructional strategies) that the helpee is comfortable with; to categorize the helper and the helpee in terms of their conceptual knowledge; and so on. Essentially, the context is used to ensure the success of the three-way dialogue between the helper, the helpee, and the help system.

Typical help contexts contain information including ontological relationships among context elements, instantiated knowledgebase about the users, inference rules pertaining to the help request, the concepts addressed by the help request, the tasks of the helpee related to the help request, the preferences of the helpee, the helper, and the system, and finally a set of instantiated plans pertaining to the current help session.

Mixed-Initiative Interactions

In a typical help-oriented interface, we find either an approach where agents instigate and control interactions through software mechanisms, or an approach where humans instigate and control interactions through direct manipulation. Mixed-Initiative refers to a flexible interaction strategy, where each agent can contribute to the task what it does best (Allen 1999). Specific mechanisms of mixed-initiatives such turn taking, grounding, confirmation, misrecognition repair, automation awareness, and attention management are used to interpret the conversants' objectives and establish a context. Recently, there has been a newly found interest among researchers in combining automation with human values – "to seek valuable synergies between the two areas of investigation…to avoid building complex reasoning machinery to patch fundamentally poor designs and metaphors…to avoid limiting designs for human-computer interaction to direct manipulation when significant power and efficiencies can be gained with automated reasoning" (Horvitz 1999).

At any one time, one conversant might have the initiative - controlling the interaction - while others work to assist it, contributing to the interaction as required. At other times, the roles can be reversed, or the conversants might be working independently, assisting each other only when specifically asked. In our view, mixed-initiative interactions are driven by conversants' relative knowledge, preferences, and task-specific toward common, partially shared, and individual goals.

We further constrain the contributions of interaction of the conversants' in terms of theory-oriented interaction models. That is, each utterance from a conversant should be interpreted and recognized within the scope of an interaction model. This allows us not only to trace mixed-initiative interactions in a theoretical framework but also verifies the validity of the theoretical foundation of the interaction model. We also contend that mixed-initiative interactions bring forth a sense of naturalness to the communication among the conversants that fosters healthy interactions among socially-oriented contexts such as 'hallway chat' or 'homework' interactions as depicted in (Rowson 2001). Importantly, mixed-initiative interactions enable a more accurate conceptualization of the relation between conversant interactions and the underlying cognitive, meta-cognitive, and socio-cognitive strategies employed by the conversants.

The advantages of using mixed-initiative interactions in problem-solving environments have been reported (Guinn 1993, Guinn 1996). Horvitz et al. (Horvitz 1999) elaborate on mixed-initiative interactions in LookOut, an interface for calendar scheduling that automatically extracts email information and updates a person's calendar. In doing so, it establishes a context around beliefs about a user's goals. In addition, LookOut also allows the user to directly manipulate the calendar. Personalized mixed-initiative interactions have also been investigated with the system learning from the contexts used by humans (Glass 2003). The modification of the system's vocabulary during user interactions could lead to highly personalized interaction mechanisms.

We have developed an ontology-oriented framework, called MI-EDNA, for mixed-initiative interactions and are currently in the process of validating the framework in the domain of reading (Shakya 2005). The framework consists of an ontology that represents information pertaining to content, learner, time, and interactions. Interactions of the conversants are automatically instantiated in the ontology. Further, patterns of specific tactics, strategies, and styles are recognized from the instantiated ontology. The recognized interactions, tactics, strategies, and styles, in an increasing order of granularity, are then mapped onto formal theoretical models (e.g., Zimmerman's 3-phase model of Self-Regulated Learning). The framework then advocates opportunities to disseminate 'well founded' prompts and other feedback mechanism to regulate the interactions among the conversants.

In this paper, we describe our reasoning in employing the MI-EDNA framework within a prototype help system named Helper's Assistant. Helper's Assistant explicitly represents contexts that can be used by all the conversants (Kumar 2001). We propose to extend the scope of the context to include variables that are relevant for mixed-initiative interactions.

Our discussion so far has focused on online help in educational systems and identified contexts as an integral component of learning technologies. In the next section we describe how contexts are employed in Helper's Assistant.

Help-context in Helper's Assistant

Helper's Assistant is a support tool for helpers in the domain of Java programming. A help-request originating from a helpee initiates the help session and the help context.

A help-context is construed as an extended representation of a help request and is a container for resources relevant for the help request. The context also includes summaries of help sessions that the helpee and the helper went through based on the feedback and commentaries the system collects from the conversants at the end of each help session.

Typically, a helpee creates a help-request that contains a question, an expected type of response, and the corresponding material (such as a piece of Java code). In addition, the help request may also map onto a set of concepts in a concept map and associated keywords. In turn, each instantiated concepts and keywords is associated with the knowledge/skill levels of the helper and the helpee.

The context also identifies a list of tasks that the helpee is currently engaged in. For instance, "assignment submission", "exams", "in-class discussions", and "quizzes", are some example tasks. Corresponding to each task, the context instantiates task models that capture the procedural knowledge associated with a task.

The help-context in Helper's Assistant also records/infers preferences of the conversants with respect to the type of help responses, the mode of help-delivery, and the form of help communication. Some example help response types are, debugging, pointer, short answer, discussion, explanation, analogy, rebuke, need more information, delay response, and provide clues. Helper can interact in three pre-defined modes in Helper's Assistant: *offline, online,* and *just-in-time*. Offline help involves asynchronous communication between the helpee and the helper using email or discussion boards as the media. Online help involves the helper sharing the helpee's workspace (or the communication channel) and helping him/her step through an on-going task. That is, the helper remains available for the duration of the task. In the just-in-time mode, help is highly specific to the question raised by the helpee and is delivered in short bursts.

The helper predominantly decides the form of the help response, either *manual* or *automated*. However, the helpee and the system can propose/negotiate their respective preferences for the consideration of the helper. In the manual form of help, communication tools and interfaces are established between the helpee and the helper so that the helper can manually (personally) deliver help using these tools and interfaces. In the automated form of help, Helper's Assistant provides the necessary help documents and help procedures to the helper who then verifies the help material and lets the system deliver the same to the helpee without any further involvement from the helper.

A help-context is created for every help session in Helper's Assistant. Each help-context is classified into a *type* in order to associate the effectiveness of a help session with the help-context. The type of the help-context is a summarization (or a signature) of help-context data deduced only from the instantiated concepts, and the preferences of the helpee, helper, and the system.

Experimental Results

An empirical study was conducted with the goal of estimating the effectiveness of help sessions with and without Helper's Assistant, among expert (teaching assistants) and peer helpers (novices). Each helper responded to four help requests, handling two requests with assistance from Helper's Assistant and two without. The help requests were derived from four buggy Java solutions to an introductory programming problem.

The help requests originated from two pre-assigned helpees. The helpees were trained to ask a specific set of questions corresponding to the four help requests. Each helper was blinded from the helpees and also from other helpers. Essentially, each helper was led to believe that the help requests and the follow-up questions were coming from real learners.

The interactions of the helper and the helpee were video taped. The mouse clicks, keyboard button pushes, and browsing patterns of the helpers across different applications were also recorded and time-stamped. In addition, each helper and helpee was asked to complete a questionnaire at the end of each help session and also provide overall feedback. The dialogues between the helper and the helpee were encoded independently by an external evaluator. Refer to (Kumar 2001) for a complete analysis of the study.

Based on the questionnaire data, as depicted in Figure 1, we observed that, both experts (56%) and novice peers (72%) are happy with the contextual information provided by Helper's Assistant. 25% of experts and 75% novice peers said that they were satisfied with the quality of pedagogical strategies. That is, novice peers preferred to have additional contextual support with respect to pedagogical strategies.

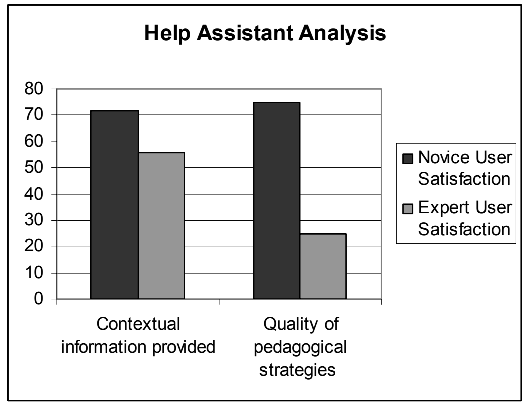

Figure 1: Helper satisfaction

The questionnaire did not probe the issue of whether experts found the pedagogical strategies less attractive because they would have preferred other strategies or they did not like strategy-oriented advice from the system.

Based on the questionnaire data, as depicted in Figure 2, we find that experts engaged in successful help sessions 88% of the time with contextual information and 85% of the time without contextual information. On the other hand, novice users had success rates of 87% and 63% respectively. Thus, novice peers find the availability of contextual information to be more useful in successfully negotiating a help session than experts did.

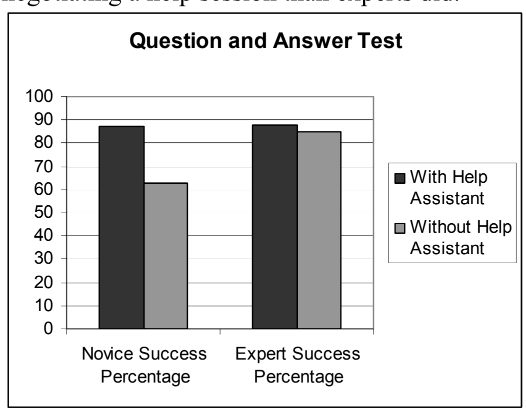

Figure 2: Helper Success Percentage

Using Wilcoxon rank-sum test (or Wilcoxon two-sample test), it was determined that there is no significant difference in the success percentage of experts between sessions with contextual information and without contextual information. On the other hand, using Wilcoxon signed-Rank test for paired observations and Wilcoxon rank-sum test, at the $\alpha = 0.1$ level, it was determined that there are differences in the medians of the success percentages of the novice peer helpers between sessions with contextual information and without contextual information.

With respect to questionnaire related to cognitive load, none of the experts felt that the contextual information provided by Helper's Assistant was too complicated to use, as opposed to, every novice peer helpers who found contextual information too complicated to use. Only one expert and one novice peer helper concurred that the context provided too much information. Two experts and three novices agreed on a midway, while all the experts and novice peer helpers said that the context provided too much information.

In general, expert helpers seemed to view the context as a reasonable tool to use with respect to complexity of use, overload of information, and ease of use in conjunction with other tools, while novice peer helpers seemed more apprehensive.

One of the key results obtained from the experiment deals with "speed of help response". A help request is assumed to have been addressed when the helper starts to "direct his/her responses" towards answering the help request. In most cases, the helpee would initiate a request, the helper would ask for some clarification, and finally the helper would start giving answers. The time interval between the start of a help request (from the helpee) to the start of the first help response utterance (from the helper) is treated as a measure of speed with which help can be offered. That is, this is the amount of time taken by the helpers to comprehend the question and initiate their responses.

All expert helpers (except one) consistently showed that they were able to provide help faster when the help system was available. Even for the odd expert helper who took more time to start providing help when the help system was made available, the difference between context establishment time is quite small (8:41 vs 9:03 minutes). All novice helpers consistently showed that they were able to provide help faster when the contextual information was made available.

Using, Wilcoxon rank-sum test, it has been observed that there is no significant difference in the average context establishment times of the expert helpers between sessions with and without contextual information. However, using the same non-parametric statistical test, at the $\alpha = 0.1$ level, it has been determined that there is significant difference in the speed of help response of the novice peer helpers between sessions with and without contextual information.

Conclusions

Based on the results from the study we emphasize the need for an explicit representation of the context for use by the conversants of the help environment. The current contextual representation of Helper's Assistant can be improvised to include variables associated with mixed-initiative interactions. For instance, the help-contexts can include which of the conversant is currently in control of the interactions and why; the help-context can advice whether a particular type, mode, or form of help interaction is beneficial; the help-context can facilitate negotiation among the conversants with respect to preferences and the degree of contextual use; the help-context can trace interactions with respect to specific theories of academic help seeking (Nelson-LeGall 1981, Newman 1994, Newman 2002) the help-context can suggest specific goals to pursue with respect to factual, procedural, and conditional knowledge; the help-context can negotiate with the helper on the effective use of working-memory; and, the help-context can estimate values for commitment, attitude, attention, and motivation of conversants.

We believe that mixed-initiative interactions, when incorporated into an online help system, allow for much greater personalization as long as the interactions are tracked and regulated with respect to specific theoretical models that employ contextual information.

Judging by how helpful the contextual information was to novice helpers, we believe that a mixed-initiative approach has good potential to improvise the help process, to increase the success rate and quality of the help sessions, and to create a much better understanding among the conversants.

Acknowledgments

This research was funded by the LearningKit project (SSHRC-INE) and the LORNET project (NSERC).

Copyright

References

Allen J. F. September 1999. Mixed-initiative interaction. *IEEE Intelligent* Systems, 14-16.

Cherry, J., Fryer, B.., Steckham, M., & Fischer, M. 1989. Do formats for presenting online help affect user performance and attitudes?.*Proceedings of the International Technical Communication Conference Washington, DC: Society of Technical Communication*. pp. RET87- RET89.

Coffee P. 1997. Office Assistant is - surprise! - useful. *PC Week, 14*(3), 49.

Collins J.A., Greer J.E., Kumar V.S., McCalla G.I., Meagher P., & Tkatch R. 1997. Inspectable user models for just-in-time workplace training. The Sixth International Conference on User Modeling (UM'97). Chia Laguna, Sardinia, Italy, 327-337.

Glass. J, Seneff. S. 2003. "Flexible and Personalizable Mixed-Initiative Dialogue Systems". *HLT-NAACL 2003 Workshop on Research Directions in Dialogue.*

Greer J.E., McCalla G.I., Cooke J., Collins J., Kumar V.S., Bishop A., & Vassileva J.I. 1998. The intelligent helpdesk: Supporting peer-help in a university course. The *International Conference on Intelligent Tutoring Systems* (ITS'98). San Antonio, TX, USA, 494-503.

Greer, J.E., McCalla, G.I., Collins, J.A., Kumar, V.S., Meagher, P. and Vassileva, J.I. 1998. Supporting Peer Help and Collaboration in Distributed Workplace Environments. *International Journal of Artificial Intelligence in Education* V. 9, 159-177.

Guinn C. March 1993. Efficient Collaborative Discourse: A Theory and its Implementation, with Alan Biermann, D. Richard Hipp, and Ronnie Smith in *ARPA Workshop on Human Language Technology*, Princeton, NJ.

Guinn C. 1996. Mechanisms for Mixed-Initiative Human-Computer Collaborative Discourse, in *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*.

Gwei G.M., & Foxley E. 1990. Towards a consultative on-line help system. *International journal of man machine studies, 32*(4), 363-383.

Hearst, M. September/October 1999. Mixed-Initiative Interaction. *IEEE Intelligent Systems*, vol. 14, no. 5, 14-23.

Horwitz E. September 1999. Uncertainty, action, and interaction: In pursuit of mixed initiative computing". *IEEE Intelligent Systems,* 17-20.

Horvitz, E., Breese, J.S., Heckerman, D., Hovel, D., Rommelse, K. 1998. The Lumiere Project: Bayesian user modeling for inferring the goals and needs of software users*. Fourteenth Conference on Uncertainty in Artificial Intelligence.* Morgan Kaufmann Publishers. 256-265. http://research.microsoft.com/~horvitz/lumiere.htm.

Houghton R.C., Jr. 1984. "Online Help Systems: A Conspectus". *Communications of the ACM*, 27, 2, 126–133.

Kumar, V. 2001. Helping the helper in peer help networks. PhD Thesis, Department of Computer Science, University of Saskatchewan, Canada.

Kumar, V. Shakya, J. Groeneboer, C. Chu, S. August 2004. Toward an ontology of teaching strategies. *Workshop Paper in ITS2004,* Brazil.

Kumar, V. 2004. An Instrument for providing formative feedback to novice programmers. *AERA 2004 Division I Education in the professions*. San Diego, USA.

Lee. H, Lee. C, Yoo. C. 1998. "A Scenario-Based Object-Oriented Methodology for Developing Hypermedia Information Systems," *hicss*, vol. 2, no. 2, 47.

Nelson-Le Gall, S. 1981. Help-seeking: An understudied problem-solving skill in children. Developmental Review, 1, 224-246.

Newman, R. 1994. Adaptive help seeking: A strategy of self-regulated learning. In D.H. Schunk & B.J. Zimmerman (Eds.), Self-regulation of learning and performance: Issues and educational applications (283-301) Hillsdale, NJ: Lawrence Erlbaum Associates.

Newman, R. Spring 2002. How self-regulated learners cope with academic difficulty: the role of adaptive help seeking. In Theory Into Practice, Vol 41, Number 2.

Rowson, J. November 2001. The Social Media Project at Hewlett Packard Laboratories. Talk at the Stanford Networking Seminar of Center for the Study of Language and Information (CSLI), Stanford University. http://netseminar.stanford.edu/sessions/2001-11-01.html.

Shakya, J., Menon S., Doherty, L., Jordanov, M., Kumar V. Accepted for publication, 2005 (in this volume). Recognizing opportunities for mixed-initiative interactions based on the principles of Self-Regulated Learning, *AAAI Fall Symposia on Mixed-Initiative Problem-Solving Assistants*, Crystal City, VA, USA.

Wise M. 1993. Using graphics in software documentation. *Journal of the society for technical communications, 40,* 677-681.