

# Evaluating a General-Purpose Adaptive Hypertext System

Chris Staff

University of Malta,  
Department of Computer Science and AI,  
Msida MSD 06, Malta

**Abstract.** We describe the evaluation process of HyperContext, a framework for general-purpose adaptive and adaptable hypertext. In particular, we are interested in users' short-term, transient, interests. We cannot make any prior assumptions about a user's interest or goal, as we do not have any prior knowledge of the user. We conducted evaluations on two aspects of HyperContext. One evaluation was completely automated, and the other involved participants. However, the availability of a test collection with value judgements would be a considerable asset for the independent and automated evaluation of adaptive hypertext systems in terms of cost, reliability of results, and repeatability of experiments.

## 1 Introduction

HyperContext is a framework for adaptive and adaptable hypertext [8], [9]. We are currently using the HyperContext framework as part of the University of Malta's contribution to the Reasoning on the Web with Rules and Semantics (REWERSE) FP6 Network of Excellence<sup>1</sup>.

HyperContext focuses on building and maintaining a short-term user model to provide adaptive navigation support. We begin a user session with an empty user model and we add to the model as a user navigates through hyperspace and interacts with the system.

A proof of concept HyperContext application has been evaluated. We had devised an evaluation strategy for HyperContext in 1999. However, due to a number of reasons, including hardware failure, the original evaluation strategy was abandoned. We eventually settled on a partially automated approach that did involve some participants, but which was less reliant on human participants.

We are satisfied that the results of the automated evaluation show that the adaptive features of HyperContext can guide users to relevant information. We feel that our automated evaluation benefited from the fact that HyperContext assumes an initially empty user model that is then populated during short interactions with the system. Part of the evaluation involved showing users a series of documents (representing a short path through hyperspace) followed by two other documents in a random sequence. One of the two documents was recommended

---

<sup>1</sup> [staff.um.edu.mt/mmon1/research/REWERSE/](http://staff.um.edu.mt/mmon1/research/REWERSE/)

by HyperContext using a user model that would have been generated had the user actually followed the path through the first 5 documents in a HyperContext hyperspace. The other document was also a recommended document, but the user model used to make the recommendation was derived in a different way. We are able to demonstrate that the second recommendation is based on a user model built on a Web-based, rather than a HyperContext-based, hyperspace. The evaluation is similar to an approach using with- and without-adaptive functionality [6], but we show that the without-adaptive functionality system is equivalent to the World Wide Web. The results of the evaluation are reported extensively in [8] and [9]. In this paper we concentrate on reporting the evaluation *process* and our opinion on its suitability for the evaluation of adaptive hypertext systems.

## 2 Objectives of the Evaluation

Before we discuss HyperContext and the evaluation strategies, we present our motivation and objectives for evaluating HyperContext. An adaptive hypertext system may use adaptive navigation techniques to guide users to relevant information in hyperspace [2]. As HyperContext utilises adaptive navigation techniques almost exclusively (there is limited support for adaptive presentation, but this was not the focus of our research), we expected that a HyperContext user would find relevant information faster than a user using a non-adaptive equivalent, as HyperContext would recommend links and paths to users, assuming that the user model accurately reflected the user's needs and requirements.

HyperContext is a general-purpose system for use in a heterogeneous information space, such as the WWW. Consequently, unlike an educational hypertext system, we cannot make certain assumptions about our users. For instance, the goal of a user of an educational system is likely to be to increase his or her knowledge of the subject contained within the system. As the domain is restricted, it is possible to pre-test or "interview" the user to initialise the user model with useful information. The users of a general-purpose hypertext system that focuses on collection of short-term information are not so helpful. A short-term user model is likely to be at its most useful when the user is navigating through territory with which he or she is unfamiliar and when the user's interest in the information is significant but transient. For instance, a user may have some task to perform and some information is required to perform that task. Although the completion of the task is dependent on obtaining the information, the user's interest in it lasts only as long as it takes to complete that aspect of the task. What motivates us is the challenge of recommending useful links (i.e., links that are likely to lead the user to relevant information) when we initially know little or nothing about the user's interests, goals, and expertise. However, motivating evaluation participants to the degree that they will search for information that they know little about but really need under evaluation conditions is hard. Either the prototype software under evaluation will need to be robust enough to use on the Web at large (in which case participants can use the system in their

own time), or a smaller Web space will be converted for use with the hypertext system (so that HTML pages, for instance, will be free from error), in which case the chances of finding adequately motivated participants is greatly reduced.

For our evaluation, we converted part of the World Wide Web Consortium's (W3C) website<sup>2</sup> to a HyperContext hyperspace. We chose the W3C site because it is about Web standards ranging from HTML to Web-HCI issues, so we reasoned that the site was designed to be easy to use, consistent, and relatively free from (HTML) errors, which would ease processing. An explanation of what is involved in the conversion is given in section 3. We also show in subsection 4.1 that without the adaptive features provided by HyperContext, the converted site is equivalent to the original Web site.

### 3 Generating a HyperContext hyperspace

In a hypertext, the same document can be the destination of many different links. Consequently, the same document may be reached along different paths. It is possible that users who reach the same document following different paths may be looking for different information, or may have reached the same document for different reasons. Such users are likely to interpret the information in the document differently, depending on the other documents in this session the user has so far read and any other knowledge and interests that the user might have. If we are to individualise link and path recommendation knowing only the user's path of traversal through hyperspace, then we need to understand how the information in the child (destination) document is related or relevant to the information in each of the child's parents.

On the Web, web pages range from short and single topic to huge, multi-topic documents. The length of a web page is not a good indicator of the number of topics it is likely to contain. Should information about all topics in a document be added to the short-term user model, in the hope that eventually the dominant topic will float to the surface? Should we use topic distillation algorithms to split up a document into its different topics, and compare each topic to the topic of the region in the parent that the user followed to reach this child? We opted for the second approach to determine the relevant terms in a document visited by the user. A document *interpretation* is a vector of term weights which partially describes a document in the context of a parent. A document has at most  $n+1$  interpretations: one for each of its  $n$  parents, and an additional one (the *context-free interpretation*), that does not decompose a document into its different topics, which is invoked if a document is accessed directly rather than by following a link to it. To convert the W3C web site to a HyperContext hyperspace we created interpretations for each (HTML) document. A link in the new adaptive hyperspace is retained if the topic distillation algorithm determines that there is sufficient similarity between the topics in the source and destination documents. The user model is updated each time the user traverses a link, using information derived from the visited document's interpretation.

---

<sup>2</sup> [www.w3.org](http://www.w3.org)

### 3.1 The User Model

The short-term user model is based on the interpretations of documents that the user has accessed during the current session. The user model is used to recommend links each time a document is accessed. A query may also be extracted from the user model and submitted to an information retrieval system to retrieve relevant interpretations if these have been previously indexed.

### 3.2 Evaluating HyperContext

As we discussed in section 2, our goal is to direct users to relevant information faster than they would be able to find it themselves, particularly when they are unfamiliar with the topic. We describe our original evaluation strategy in section 4. In section 5, we describe the actual strategy we used to evaluate HyperContext. In this paper, we concentrate on the evaluation *process*. The evaluation results are discussed in detail elsewhere [8], [9].

## 4 Evaluation Strategy 1

The empirical study that we had originally planned was to involve three groups of six participants each. Of the 18 participants, 6 each were previously judged to be novice, intermediate, and advanced information seekers. The initial study involved 36 participants who were set 15 general knowledge information seeking tasks. They were allowed to use any information source (search engine, web directory, their own memory) they liked, but had to indicate if they already knew the answer. For each task, the student had to write down a URL containing the answer (or URLs, if the answer spanned a number of web pages). The information seeking tasks were pre-tested to ensure that the answers were available on the Web.

Each participant's performance for each task was compared to the average time to perform each task (from among those participants who did not already know the answer). Participants who generally arrived at a solution faster than average were considered advanced information seekers, those who were generally much slower at finding information were considered to be novice, and the others were considered intermediate. 6 people were to be randomly selected from each group to participate in the HyperContext evaluation.

A HyperContext Evaluation group was to consist of two novice, two intermediate, and two advanced information seekers. Each group would have an identical set of tasks to perform. The tasks were designed to find technical information, rather than general knowledge as used in the experiment to classify participants. One group would act as the control group, the second and third groups would both use a HyperContext-enabled version of the W3C web site, but the algorithm used to construct the user model would be different. Once again, the performance of the two HyperContext-enabled groups would be compared to the performance of the control group, where we can show that the control group would have used

the equivalent of the W3C web site. Each group would have access to the same information search and retrieval system. The control group would have access to an index generated from the original, unmodified documents, whereas the other groups would have access to an index that also contained an index of document interpretations (document interpretations are discussed in section 3).

#### **4.1 Is a without-adaptation HyperContext equivalent to the Web?**

The HyperContext hyperspace created from the W3C web site for use in the evaluation (section 3) can be considered equivalent to the original W3C web site if adaptivity is disabled. By default, the context-free interpretation of a document consists of a vector of term weights for all terms that occur in the document, rather than just those terms that are considered relevant to the parent, when a link is followed. In the disabled version of HyperContext, all link traversals invoke the context-free interpretation, so the interpretation of the document is the same regardless of how the document is accessed. This behaviour is equivalent to the behaviour on the Web. Regardless of how any page is accessed, normally there is absolutely no difference in or about the page that was accessed.

## **5 Evaluation Strategy 2**

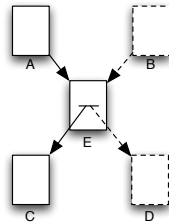
Due to a number of unfortunate incidents, including hardware failure resulting in total data loss, and looming deadlines, the intended strategy outlined in section 4 never progressed beyond the first stage of classifying participants as novice, intermediate, and advanced information seekers. By the time the HyperContext hypertext and related data were recovered, there was simply not enough time to re-run the original classification of participants (because their information seeking skills were bound to have changed over time [3]), conduct the rest of the evaluation and analyse the results. Instead, we decided to separate the evaluation of some of the functionality from the evaluation requiring user participation [5]. We developed one completely automated experiment to test our hypothesis about the improved ability to locate relevant information in a HyperContext hyperspace. A second experiment required anonymous Web-based participation from users to judge whether documents recommended by HyperContext were relevant to information they had read on a pre-determined path through a HyperContext hyperspace.

### **5.1 Locating relevant information**

The number of links on a page, coupled with the lack of a link semantics in HTML increases cognitive overhead. A user must decide whether or not to follow a link. Adaptive educational hypertext systems may make use of visual link adaptation to indicate that a link may be followed with profit, or should not yet be followed, e.g., [11]. Alternatively, forms of link hiding [2] may be used, in which users are discouraged from following links unlikely to lead to relevant information. In

either case, this is a form of hypertext partitioning - separating the non-relevant parts of hyperspace from the relevant.

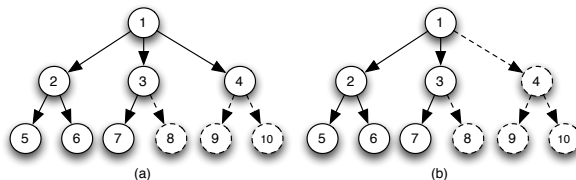
In HyperContext, a user visiting a document actually visits an interpretation of that document. In Section 3 we explained that an interpretation is a vector of term weights, and that different interpretations of the same document may have different vectors of term weights. For instance, in one such vector, some term  $t_n$  may have weight  $w_x$ . In another interpretation of the same document, the same term may have the same weight, or a completely different weight, depending on how significant the term is to the context of the topic of that document's parent. Interpretations are slightly more complex, however. One interpretation of a document may have link anchors which may or may not be active in other interpretations (a form of link hiding). Additionally, even if the same link is active in several interpretations of the same document, the destination of the link may change depending on the interpretation (figure 1). In this way we are able to partition a HyperContext hyperspace, potentially separating the non-relevant from the relevant.



**Fig. 1.** Link in doc E leads to C if entered from A, and to D if entered from B.

To determine if multiple interpretations of information can adequately partition a hyperspace so that a user can be led to relevant information, we count the number of nodes that must be visited starting from some arbitrary start node until we reach a relevant node. A relevant node is just some randomly selected node that is at least 2 link traversals away from the start node. We compared two adaptive solutions to two non-adaptive solutions, measuring overall performance and the performance of each approach as the path length grew. The adaptive solutions were based on a HyperContext enabled converted W3C hyperspace, and the non-adaptive solutions were based on the original W3C web site. The premise is that the optimal solution is one that finds the shortest path between the start node and the target, relevant, node. The least optimal solution is likely to be based on a breadth-first or depth-first brute force search (depending on the “shape” of the hypertext graph), essentially following the links in the order of least likely to lead to the target node. For this experiment we traversed the links in the order they occurred in a document, using a hybrid approach. We process nodes breadth-first until we encounter the target node. We then prune the graph

of accessed nodes, eliminating all visited nodes to the right of the shortest path between the start node and the target node (figure 2). This is the equivalent of a depth-first search guarded by the known depth of the target node. If the best link to follow always happens to be the first link in a document, then this approach will give results similar to the optimal solution. However, unless the best link is always the last one in a document, then this approach will give better results than the least optimal solution, because nodes which did not need to be visited will not be counted. This approach yielded paths of maximum length 5 (four link traversals from the root).

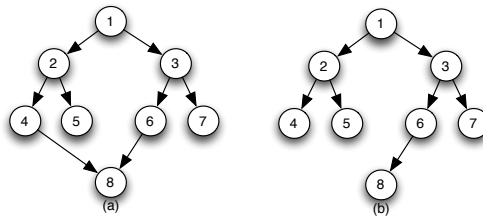


**Fig. 2.** Node 7 is the target node. (a) Solid nodes are visited in breadth-first search; (b) hybrid depth-first marks node 4 as unvisited.

An algorithm that partitions the hyperspace may decrease the span of the graph, and hence improves the speed with which a target node can be reached, even when a brute-force approach is taken. The efficiency may be decreased if a relevant node is made either unreachable or reachable by a longer traversal of the graph if the hyperspace is partitioned badly (figure 3). In either case (adaptive or non-adaptive) the efficiency may be further improved by introducing a link ordering algorithm that ranks links in a document according to the likelihood that they will lead to the target node. The link ordering algorithm compares the current node’s children (a lookahead of 1) to the target node. Links in the current document are traversed in the order of degree of similarity between the link destination and the target node. In the experiments with the adaptive version of the hypertext, the interpretation of each child (section 3) is used by the algorithm, rather than the context-free interpretation of the child used in the non-adaptive version.

## 5.2 Evaluating Document Recommendation

In the second part of the evaluation, we prepared a number of paths through hyperspace that all involved exactly four link traversals (for consistency with the maximum path length reported in subsection 5.1) through different documents. If a document was re-visited on a path, the path was not selected for the experiment. Two user models were maintained. We assumed that the first document on the path was the root of the path, and that both user models were empty at this point. Each user model was updated following a link traversal to



**Fig. 3.** Partitioned hypertext: (a) before, and (b) after. Node 8 is no longer reachable from node 4 in (b) and so may take longer to reach.

the next document on the path. On reaching the last (fifth) document on the path, two queries were generated, one from each user model, and submitted to our search engine. The first document recommended by each user model was noted. Eventually, participants were asked to give relevance judgements about each recommended document having first read all documents on the path.

A term weight vector based on the interpretation of each visited document on a path is used to update the first user model ( $UM_{adaptive}$ ). For the second user model ( $UM_{control}$ ), the context-free interpretation of the document is used<sup>3</sup>. If both user models recommended the same document, the path was considered inapplicable for evaluation purposes and was discarded. Eleven conforming paths of length five were randomly selected. The documents on the path and the documents recommended by each user model were placed on-line and hosted by a Web server for 25 days. Members of staff in the Department of Computer Science and AI at the University of Malta and its student population were invited via e-mail to participate in the on-line evaluation. Participation was totally anonymous and could be carried at the participant's leisure from a location of their choice. Participants were asked to read each of the first five documents in a path in the order they were displayed. They were then shown two recommended documents (one after the other) and asked to give a relevance judgement about each.

We used a 4-scale of relevance judgements (highly relevant, quite relevant, quite non-relevant, highly non-relevant), rather than the two (relevant, not relevant) normally used [10], because we expected both user models to make recommendations of at least slightly relevant documents. Participants were not told the order in which recommended documents would be displayed. They did not know which document was recommended by  $UM_{adaptive}$  and which was recommended by  $UM_{control}$ . The sequence was set randomly.

### 5.3 Summary of Results

The results of the evaluation are reported extensively in [9] and summarised in [8]. To locate relevant information we measured the difference between the best

<sup>3</sup> This is the equivalent of the Web version of the document (section 4.1).



case scenario (the shortest path between two nodes), the worst case (the longest path assuming that we know the level depth of the target node), and the adaptive solutions. The adaptive solutions outperformed the non-adaptive ones as path length increased. If the target node was 3 or 4 link traversals from root, then the adaptive solutions found the target node having visited less intermediate nodes than the non-adaptive approaches. This performance was reversed for target nodes that were up to 2 links traversals away from the root.

For the second part of the evaluation, two user models were used to recommend documents to users using an adaptive and a non-adaptive approach respectively. At face value, documents recommended by the non-adaptive approach were considered more relevant than those recommended by the adaptive approach. However, if time spent reading a document is an indication that a document is skim read or read closely (deep read), then readers tended to consider relevant the document recommend by the adaptive approach when the documents were deep read, and those recommended by the non-adaptive approach if the document was skim read. However, this is an assumption because although we measured the amount of time spent reading each document on a path users were not asked to confirm whether they skim or deep read the documents.

## 6 Conclusion

One main and significant difference between general-purpose adaptive hypertext systems, like HyperContext, and adaptive educational hypertext systems is that our evaluation participants did not necessarily have any motivation to read about or learn about the information contained in our hyperspace (Web standards). In educational hypertexts, there may be more scope for finding participants who are interested in learning what the system is teaching. We feel that HyperContext would have benefited from evaluation by participants who use it to guide their search for information that they are motivated to obtain. However, setting up such experiments can be complex and expensive [4]. For example, the Alberta Ingenuity Centre for Machine Learning pays an honorarium to Web-based participants in the evaluation of LILAC<sup>4</sup>.

Creating test collections with value judgements for adaptive hypertext systems may make the results of automated evaluation more reliable and comparable, as has been the case with information retrieval and systems for some decades [1]. Perhaps the most common criticism of this approach, and one that could also effect adaptive hypertext systems, and not merely because some, like HyperContext, make use of information retrieval systems to make recommendations, is that *relevance* is highly subjective. The Text Retrieval Conference uses “pooling” to set relevance judgements for documents in test collections [10], [7].

We automated some of the evaluation process for HyperContext. We selected the algorithm for updating the user model, and we used a simple topic distillation algorithm to create interpretations of documents based on each of their

---

<sup>4</sup> [www.web-ic.com/lilac/honorarium.html](http://www.web-ic.com/lilac/honorarium.html)

parents to partition hyperspace so that we can more quickly locate a target document presumed to contain relevant information. Of course, this automated experiment alone was insufficient to conclude that users would actually find the recommended documents relevant, so we then invited participants to provide relevance judgements for documents that were recommended after the participants had read 5 documents on a path through a converted W3C web site.

We use a short-term user model that is initially empty to collect information about a user's interests as the user navigates through hyperspace. This is the only way in which the user model can be updated. If a user is not permitted to use a search engine to locate information, or to jump directly to pages using their URL, or to directly edit the user model, but can only follow paths through the information space, then the user model of all users following the same path will be updated in the same way, and the same recommendations will be made. If we can also know in advance which links and documents should be recommended at each stage, then it should be possible to create a test collection with relevance judgements that can then be used for automated evaluation.

## References

1. Baeza-Yates, R. and Ribeiro-Neto, B.: *Modern Information Retrieval*, Chapter 3: Retrieval Evaluation (1999) Addison Wesley:US
2. Brusilovsky, P.: *Methods and techniques of adaptive hypermedia*. *Adaptive Hypertext and Hypermedia* (1998) 1–43. Kluwer Academic Publishers:Netherlands.
3. Chin, D. N.: *Empirical Evaluation of User Models and User-Adapted Systems*. *User Modeling and User-Adapted Interaction*, v. 11, n1-2 (2001) 181–194
4. Del Messier, F., and Ricci, F.: *Understanding Recommender Systems: Experimental Evaluation Challenges*. Weibelzahl, S., and Paramythis, A. (eds) *Proceedings of the Second Workshop on Empirical Evaluation of Adaptive Systems in conjunction with UM2003* (2003) 31–40.
5. Herder, E.: *Utility-Based Evaluation of Adaptive Systems*. S. Weibelzahl and A. Paramythis (eds), *Workshop on Empirical Evaluation of Adaptive Systems User Modeling 2003* (2003) 25–30
6. Höök, K.: *Evaluating the utility and usability of an adaptive hypermedia system*. *Proceedings of the 2nd international conference on Intelligent user interfaces*, Orlando, Florida, United States (1997) 179–186
7. Smeaton, A. F., and Harman, D.: *The TREC Experiments and their impact on Europe*. *Journal of Information Systems* (1997)
8. Staff, C. D.: *The HyperContext framework for Adaptive Hypertext*. *Proceedings of the Thirteenth ACM conference on Hypertext and Hypermedia*, College Park, Maryland, USA (2002) 11–20
9. Staff, C. D.: *HyperContext: A Framework for Adaptive and Adaptable Hypertext* (2001) PhD thesis, University of Sussex.
10. Voorhees, E.: *Overview of TREC 2003*. *Proceedings of the Twelfth Text REtrieval Conference*. (2003)
11. Weber, G. and Brusilovsky, P.: *ELM-ART: An adaptive versatile system for Web-based instruction*. *International Journal of Artificial Intelligence in Education* 12 (4), Special Issue on Adaptive and Intelligent Web-based Educational Systems (2001) 351-384.