Information Retrieval from Hypertext Using Dynamically Planned Guided Tours

Catherine Guinan and Alan F. Smeaton
School of Computer Applications, Dublin City University
Glasnevin, Dublin 9, Ireland
e-mail: smeaton@dcu.ie

Abstract

In using any hypertext system a user will encounter many technical problems which have been well-documented in the literature. Two of the more serious problems with using hypertext are user disorientation and the retrieval of information. Another less often addressed problem is that of the logical sequencing of nodes. In the work reported in this paper we address these three problems by combining Hammond and Allinson's guided tour metaphor and Frisse's information retrieval techniques to dynamically create guided tours for users in direct response to a user's query. One of the features of our method is that we take advantage of typing of information links in the hypertext to generate a tour which has a judicious sequencing of nodes rather than a simple presentation of hypertext nodes in order of similarity to the user's query. Our method was empirically tested on a population of 125 users who generated a total 973 individual tours and all user actions and responses to questions were logged. The results of this evaluation are presented in this paper.

Acknowledgement: CG would like to acknowledge the support of Telecom Éireann and the Irish-American partnership Scholars program for supporting her in this research. All correspondence regarding this work should be addressed to AFS.

1 Introduction

In using any hypertext system an end-user or browser will encounter many technical problems which have been well-documented elsewhere [10]. Two of the more serious of these problems are disorientation and the quality of the information retrieved by the user from the hypertext. In disorientation a user may effectively get lost in the hypertext and not be able to return to a node they previously visited or follow a trail of nodes they found interesting. Related to this is the problem of actually retrieving relevant information from a hypertext. If a user is simply browsing around a hypertext with the sole purpose of finding any interesting or potentially useful information then the browsing metaphor which hypertext encourages is adequate. If, however, a user has a more definite purpose and seeks a specific piece of information from a hypertext then unless the hypertext has been very well-engineered then the user may not be able to retrieve relevant information [16].

In addition to the two problems with using hypertext mentioned above another, less often addressed problem is that of the logical sequencing of nodes as they are presented in a hypertext browse. An end-user of a hypertext has complete control of the sequence of nodes presented to him/her when browsing which is a facet of hypertext with much appeal which can also cause problems as that user tries to assimilate information contained in the nodes that have been viewed. If the logical sequence or the order of node presentation is askew or unbalanced in terms of information content then garnering an overview of the overall information presented will be made more difficult, even if the nodes that have been viewed are all relevant to the user's information requirements. For example, if a user is browsing a hypertext on database topics then it is much more sensible for a user to examine overview and introduction nodes before nodes detailing examples and syntax of SQL !

In this paper we explore a method which we believe directly addresses the three problems with browsing through hypertext that we have mentioned above, disorientation, information retrieval and the logical sequencing of nodes. We propose a method for dynamically generating a guided tour through a hypertext in response to a user's query which will plan a tour through a hypertext covering all the nodes the system believes relevant to the user's information need as expressed in a natural language query, which eases disorientation problems by using the guided tour metaphor first proposed by Hammond and Allinson [8] and which takes advantage of typing of information links in the hypertext to present the nodes on the tour in a logical fashion. Effectively we extend the guided tour metaphor by including the information retrieval features proposed by Frisse [5] and we add in logical sequencing of nodes on the tour to complete the process.

The rest of this paper is structured as follows. In the next section we give a brief overview of related work on guided tours, incorporating information retrieval into hypertext browsing and we look at how typing of links and/or nodes has been used in other work. The subsequent section details the mechanism we have devised for dynamically planning tours. In section 4 we give an outline of our experimental environment and in section 5 we present some of our results. A concluding section summarises our work and gives some ideas for future work.

2 Retrieving Information from Hypertext

The metaphor of providing a guided tour through a hypertext to ease problems of disorientation was first proposed by Hammond and Allinson [8] and also used by Trigg [15] in the NoteCards system. In this approach, guided tours were created by the hypertext author and were fixed, static features of the hypertext. Problems of authoring such guided tours have been well- documented in [9] and the notion of a guided tour has been extended and refined numerous times. For example, Zellweger has introduced the idea of scripted documents which are more complex than static guided tours in that scripted documents can have conditional and programmable paths, automated playback and active entries [18] but the common denominator is the approach of offering to guide a user through a hypertext if the user wants to follow the guide.

Traditional information retrieval functionality of ranking a set of documents or texts in response to a users query as described in [12] has been around for a long time and is reasonably well understood. Hypertext-based retrieval of information, effectively by browsing around an information space following authored links, is comparatively recent in popularity. The former is useful for querying, for satisfying user information needs where users know exactly what they are looking for while the latter is more effective where users don't know exactly what they are looking for and are browsing [16]. In between the two extremes there are user information needs which have elements of both of the above and because of this there have been some previous attempts at combining information retrieval and hypertext.

Methods for searching for information in a hypertext have been developed by Frisse [5] who used statistically-based information retrieval techniques to choose a starting node in a hypertext from which users could commence a browsing session. The computation to find the best starting node is based on a user's initial query to the system and recommends a starting point for a user but does not then provide any further guidence to that user in finding relevant information in the hypertext. Dunlop and van Rijsbergen [3] have developed a method for retrieving non- textual information from a hypertext, again using statistically-based approaches.

Coombs [2] describes a system for searching through a hypertext network using natural language queries and boolean based retrieval. Searching in this system can also be restricted to a particular area of the network instead of encompassing the entire information base. These ideas have been implemented and tested in IRIS Intermedia. α–Trellis by Furuta and Stotts [6] is a prototype hypertext browsing and authoring system which incorporates the users browsing experience into the nodes and links structure of hypertext. Petri nets are employed to facilitate this two way specification. Finally, Shepherd et al [14] have developed a system of transient hypergraphs for their citation network whereby the entire hypertext is represented as a hypergraph or a labelled directed graph. Hypergraphs are generated dynamically in response to a users query and are discarded at the end of the query session. Different perspectives of the hypertext can be gained by the user during each query session.

In order to increase the richness of knowledge representation in a hypertext the nodes and/or the links may be typed. The term *type* is somewhat overloaded as it can mean many things but in a hypertext context is means a classification of nodes and/or links into one of a pre-defined set of categories. Typing has become quite popular in hypertext systems because of the fact that in general it substantially enhances the quality of the representation of information in the hypertext. Typing has been used in systems like gIBIS, PHI, NEPTUNE, NoteCards and CONCORDE [1]. Ideally, a hypertext system incorporating typing should allow author-defined types as well as having a generic set of types common to most applications. Typing of nodes and/or links in the hypertext may subsequently prove useful in retrieval from hypertext and this is an active area of research in the field.

3 Dynamically Planning Guided Tours

The dynamic aspect of our guided tour comes from the fact that tour creation is postponed until the user enters a natural language search query. This differs from the static tour described earlier in which the hypertext author pre-defines all tours before any user has formulated any information requirements. The guided tour is also logically *planned* in such a way that attention is paid to the types on the information links connecting the individual nodes in the hypertext before a route through those selected nodes is computed.

The method used for selecting the nodes to be included in the tour is based upon the work done by Frisse described in [5]. However unlike Frisse's application, the hypertext we have used does not have a hierarchic topology but more of a network. Propagation of weights upwards as Frisse has done can not therefore apply directly in our case. However, as we shall see, the weights assigned to nodes do depend on the number of immediate neighbours and their weights so the concept of an extrinsic weight as proposed by Frisse still applies.

Upon entering the system the user is asked to input a natural language query from which all stopwords are removed and the remaining words stemmed using a standard algorithm from information retrieval research [11]. An inverse document frequency (IDF) weight is then calculated for each query term which assigns a higher weight to terms which occur less frequently throughout the collection. Each node in the hypertext is given a score which is initially the product of the term frequency or number of times a term occurs in that node, and the IDF weight ($tf * IDF$), for each of the query terms. This term weighting strategy used to assign scores to nodes has been shown to have excellent performance in terms of effectiveness for information retrieval and is also relatively undemanding in terms of computing resources.

The number of connections from each node is also considered to be important in determining which hypertext nodes to include in the tour. Not only must each selected node be relevant to the user query but it must also be well connected to the other nodes in the hypertext to allow the user the option of deviating from the tour. The *goodness of an area* as opposed to the *goodness of a node* is considered to be more important for retrieval purposes. For example, if the weighting mechanism chose as the highest scoring node one which had no connections to any other node in the hypertext this would not be deemed as an appropriate starting point in a guided tour compared to a node which scored less but was immediately connected to many other medium scoring nodes. Therefore, some method of combining a node's score with its neighbour's scores had to be used.

One method of incorporating a neighbours contents into each node as used by Frisse [5] is by calculating the average score of the neighbouring nodes and adding this figure to the current node to give a measure of the utility of each node. This procedure should be repeated for all nodes in the hypertext. In this way, some provision is made for calculating the *goodness of an area* as opposed to *goodness* of each individual node. Formally, the method of measuring the utility of a node can be given as

$$I_i = W_i + \frac{\sum_{j=1}^{n} W_j}{n}$$

where W_i is the $tf * IDF$ weight of node i and W_j is the $tf * IDF$ weight of one of the n nodes *connected* to node i via an authored information link. Having calculated a utility score for each node based on within-node query term frequency, inverse document frequency and contents of linked neighbours, one final adjustment is made to the node scores. In the hypertext we have used each node consists of a short descriptive title plus a body of text. If a node contains some of the query terms in its title then it is more likely that the contents of that node relate specifically to the search query and so should be more certain of inclusion in the tour. Increasing the weight is one way of ensuring that the node is included in the top ranked nodes.

Once each of the nodes in the hypertext have been assigned a utility score based on the user's query, it is necessary to establish a cut-off point (μ) below which all nodes whose utility score is less than are discounted from inclusion in the tour. All nodes whose weight is greater than or equal to this cut-off or threshold value would be selected for inclusion in the tour.

Initially it was not known what would be the ideal threshold value so this was one of the parameters whose optimal value we determined empirically. The threshold value would also vary for each user query as the strengths of the query-node similarities would vary between queries depending on how close they were to the information in the hypertext. In our experiments reported later, every time a tour is generated for a user, a threshold value was randomly chosen as some fraction of the utility score of the highest-scored node. The fractions used initially came from the set {0.1, 0.2, 0.3, 0.4, 0.5, 0.6}. If, for example, $\mu = 0.3$ was randomly selected, then before a node is selected for inclusion in the guided tour its utility score must be at least 30% of the score of the highest scored node. Depending on which value from the set is selected the actual number of nodes in the tour can vary greatly so the threshold value chosen actually controls tour length.

At this stage, the nodes for the guided tour have been dynamically chosen in response to a user's query and some form of logical ordering has to be imposed on the selected tour nodes. We decided to exploit the relationships between the nodes to plan an ordered path. As part of the hypertext authoring process, the link types between nodes as shown in Table 1 had been used.

Type-1	is_a a_kind_of is_contained_in
Type2	consists_of contains preceeds
Type3	has facilitates refers_to is_related_to

Table 1. Node-node Link Types

A precedence order was established among link types. Higher precedence is given to those link types which define a clear parent-child relationship thereby indicating a distinct hierarchical association between the nodes. Link types in this category are of type-1. Second in order of precedence are type-2 links which indicate that the nodes from which these links emanate should logically precede the destination node where both nodes are to be shown to a user. Lastly, checks are made for the remaining link types, type-3 links.

Once the nodes selected for inclusion in the tour have been ranked by their utility weights, the highest scoring node is examined for links which indicate that it is the child of a parent node also in the tour. These are type-1 links. If a link of this type is present it is followed and the destination node is then examined in the same way provided it is included in the tour. This process continues until the node highest in the tour hierarchy is found. This node becomes the first to be displayed in the tour. It is then examined for type-2 links which indicate that a definite child node exists which should be displayed next. If none exists then type-3 nodes are examined in the following order, ***has, facilitates, refers_to***, and then ***is_related_to***. dependThis procedure is followed until every node selected for inclusion in the tour has been positioned correctly relative to each other node in the tour. This positioning is facilitated by means of the link types as mentioned above.

In the next section we shall describe the experiments we ran to determine the optimal value of μ the threshold value for determining inclusion of a node in the tour.

4 Experimental Details

The hypertext browser we developed runs on SUN SPARC workstations using OpenWindows version 2. We used the ORACLE R.DBMS for storage management and the system was written in C with embedded SQL commands. The hypertext we used consisted of 551 nodes of information about the undergraduate course on Databases at Dublin City University. Among these nodes there are a total of 1566 typed links. These links were created manually using the ten distinct link types given earlier which were deemed to encompass the most frequent and important relationships existing between the various nodes. Some of these link types may be applicable to all hypertexts but some will be application dependant. Depending on the contents of each individual hypertext, the author will have to determine the relationships between the various nodes as there are no set guidelines on the creation and typing of links. Gray and Shasha devised links of type *example, critique, counter-argument, compared-with* and *continuation* for their NetBook system [7]. Yet another author might formulate different link types for the same system.

The browser we developed is meant to be used in tandem with the Databases lecture course. The user population consisted of 125 undergraduate students who used the system over a period of three months. Their usage of the browser was broken up into three separate batches or assignments. An initial assignment for students was used for familiarisation in which the user became acquainted with the material covered in the database and with how to actually use the browser tool. A user session consists of everything a user does *in one sitting*, including the search query and all the button presses made. A log file was maintained which contains details of every user action during each session. The familiarisation assignment was not logged and the results presented here are for what we term the first and second assigments. The log file details include username, the user's query, tour contents, button presses and their timestamps, etc.

At the end of each session, users were asked a series of questions about the generated tours to give us some feedback on what they thought of the guided tours. These questions asked about users' perceptions of tour length, information coverage and coherence and the responses were analysed. Responses to questions were mandatory in our browser in order to correctly exit the system.

Different tours, even for the same query, were generated by varying the threshold value at which nodes qualify for inclusion in the tour. In assessing and comparing the generated tours, three distinct characteristics can be identified.

- *Tour Length:* This deals with the number of nodes which have been selected for inclusion in the tour. Depending on the threshold value and on the search query the tour length can vary greatly. The more general is the users query, the wider the range of nodes which are eligible for inclusion in the tour while the more narrow or specific the query, the fewer nodes will qualify. Depending on the threshold value μ, which is randomly selected, the length of the tour can vary from consisting of only a few nodes to as many as 50. If a high threshold, say $\mu = 0.5$, is chosen, then fewer nodes will meet the entry criteria of being at least 50% of the highest scoring node than if $\mu = 0.1$ in which case many more nodes will be selected as their utility score only has to be at least 10% of the highest scoring node. The same query entered by two different users at the same time can result in two different tours.

- *Information Coverage:* This characteristic of a guided tour also deals with long versus short tours but is more concerned with the relevance of the retrieved information. For each query, there exists a certain set of relevant nodes and a larger set of irrelevant nodes. The ideal hypertext system would select for inclusion in the guided tour only those nodes which belong in the relevant set. Unfortunately, statistically-based information retrieval techniques like $tf * IDF$ weighting are not sophisticated enough to guarantee this [13]. Obviously, the threshold value influences the information coverage also as the longer the tour the more general will be the information coverage and the shorter a tour, the more specific will be the information coverage.

- *Connectivity:* This characteristic deals with how well-connected or how fragmented is the presentation order of the nodes in the guided tour. The connectivity of a tour can be measured in terms of the existence of authored information links between tour nodes. A tour is considered to be well-connected if each node in the tour is linked via an authored link to the node that follows it on the tour. The hypertext author is the only person who can explicitly define an information link. A tour is considered to be fully fragmented if each node in the tour has no information links to any of its neighbours on the tour. The well-connected tour is the type considered to be ideal for a hypertext used as a learning aid. In this case, each node leads logically to the next in line and no concept is discussed in detail until it has been fully explained in a previous node. The fragmented type of tour consists of nodes plucked from different areas of the hypertext, many of which may have no logical ordering but appear in a random fashion. Again the threshold value can affect this characteristic. If the tour is very long more nodes are included and so the chances of having a highly connected tour are greater. Conversely, if the tour is short, no avenue of discussion may be followed too deeply thus leading to a more fragmented type of tour. Theoretically, changing threshold values should have this effect but in reality, the situation may be very different.

To date, no tried and tested method of evaluating hypertext systems has been devised and so our results and analysis will be mainly based on observation of user behaviour and on the answers given to the questions asked at the end of the user session. The increased cognitive complexity of using hypertext systems makes them more difficult to evaluate than information retrieval systems [17]. Two of the characteristics of guided tours which we shall examine, tour length and information coverage will be evaluated by inspecting the users replies to the questions and so will be subject to human error and bias. However, it can be determined by examining these results if the tours have been useful to end users.

5 Experimental Results

There were 580 logged user sessions or tours generated in the first assignment and 393 in the second assignment. Each student on the course was expected to run at least 4 tours in each of the two assignments being logged though some students actually did more, some did less and some did none. Figures such as number of nodes visited and session length did not vary much across the two assignments. In both cases the pattern was that of an exponentially decreasing distribution, i.e. many users visited a few nodes and few users visited many nodes, a result which has been found elsewhere. The details are given below:

	Assign 1	Assign 2
Avg. total nodes seen	23.9	18.4
Avg. time per session	11.9 m	9.8 m

Table 2. Use of browser

Tour length was the first of the results we examined and the users answers to the questions in both assignments differ and are given in Table 3 with the responses *far too long, too long, just right, too short* and *far too short*. The averaged tour length for different values of μ are given in Table 4.

	FTL	TL	JR	TS	FTS
Assign 1	5%	25%	57%	10%	3%
Assign 2	3%	10%	63%	19%	5%

Table 3. User responses to questions on tour length

μ	Assign 1	Assign 2
0.1	34.6	–
0.2	24	–
0.3	16.4	15.6
0.4	12	11.7
0.5	9.1	7.9
0.6	–	5.4

Table 4. Avg. tour lengths for threshold values

In the first assignment, when μ is in the range 0.1 to 0.5, a sizable percentage (43%) of users replied that the tour length was not satisfactory, it was either too long or too short. In the second assignment, 37% of users replied that tour length was unsatisfactory. This may not seem like a large difference in unsatisfactory tour lengths between the two assignments but the important feature to note is how these figures are made up. For example, in assignment 1, 30% of users replied that the tour length was too long and only 13% said that it was too short. In assignment 2, only 13% of users replied that the tour length was too long whereas 24% said that the tour was too short. It could be argued that users were even more familiar with the browser tool in assignment 2 but the shift in pattern could not have been caused by users familiarity with the hypertext as users queries and hence tours, differed across assignments. We believe that the shift could only be caused by the changed set of values for μ as this was the only parameter to be modified between assignment 1 and assignment 2. After the first assignment, tours were made shorter by removing $\mu = 0.1$ and $\mu = 0.2$ from the set of values actually used and $\mu = 0.6$ was added. In relation to tour length, the most suitable threshold value seems to lie between $\mu = 0.3$ and $\mu = 0.4$ but until the tour generation algorithm is tested against more hypertexts this value cannot be regarded as universal.

Information coverage is based on what the user deems to be relevant or irrelevant on a tour with regard to their particular query. Trying to measure this characteristic exactly would be very difficult as it would be necessary to select all the nodes in the hypertext which were relevant to every user query. This is the same problem as faced in typical information retrieval experimentation where exhaustive relevance judgements on a collection of texts are ideal but not always available [12]. For this reason, it was decided to gauge information coverage based on user's replies to questions about the proportion of relevant nodes on the tour. A long tour is not necessarily better as it may contain a lot of information which is extraneous to the query thereby forcing the user to sift through all of the material in order to find the appropriate nodes. Likewise, even though a short tour may contain only those nodes which are relevant it may also omit several suitable nodes, thereby compelling the user to browse in order to find all the information relating to their query. Our task is to find a suitable mid-point between these two extremes. One pattern to notice in examining the user's replies to the question on information coverage is that as increases so does the number of users who replied that the tour nodes were highly relevant. Similarly, as μ increases, so does the number of users who replied that nodes visited outside the tour were relevant. This pattern adheres to the principle that as tours get shorter less irrelevant nodes are included in the tour. However, it also appears that some relevant nodes are being excluded from the tour. It seems that high threshold values do not lead to tours which contain enough information to satisfy the user's query. Likewise, when the tours were long, $\mu \epsilon \{0.1, 0.2\}$, the proportion of users replying that a high percentage of tour nodes were relevant was lower than when a high threshold value, $\mu \epsilon \{0.5, 0.6\}$, was used. By examining the results, it seems that when $\mu = 0.3$ the information coverage is enough to satisfy the users query but not too much that they are laden with material or too little that they have to go in search of it themselves.

In order to measure a third characteristic of guided tours, their connectivity, a scoring algorithm was devised which, using the link types between adjacent tour nodes, calculated a connectivity score for each generated tour in both assignments. The link types which were used to connect the nodes in the database were divided into three groups which represent how important each link type is deemed to be in providing a connected and logical path through the network of nodes. Most important are the link types which determine where each node should appear in the tour relative to its neighbours. Links in this category are *is_a, a_kind_of, consists_of* and *contains.* If two nodes are connected by any of these links then it is immediately apparent which one should come first in a tour. Second in terms of link strength are links which more loosely determine an order of precedence which include *precedes, has* and *facilitates.* Finaly are those links in which no attempt is made to establish the order in which links should be displayed.

Having thus ascertained the strengths of the various link types in a tour we then gave each generated tour a score based on its coherence. The formula to do this is based on assigning weights to each of the link categories; 3 for the first category, 2 for the second and 1 for the third. When all links in each tour have been taken into consideration the tour score is normalised by dividing by the total number of links in the tour. The following are the scores assigned to tours in each assignment. The scores are divided up into bands of 0.3, e.g. 0 to 0.3, 0.3 to 0.6, etc. and are given in Table 5.

μ value	Assign 1	Assign 2
0.0 – 0.3	8%	16%
0.3 – 0.6	28%	28%
0.6 – 0.9	43%	33%
0.9 – 1.3	15%	15%
1.3 – 1.6	5%	4%
1.6 – 3.0	1%	4%

Table 5. Connectivity scores for tours

In both assignments, the connectivity of most of the tours fell into the range 0.6 to 0.9. When these scores are compared with a control group of tours, we can see how ordering or sequencing the nodes on the tour using link types as outlined earlier, affects the connectivity of the tours. The control group was used to determine the connectivity scores which would be calculated without the logical ordering of nodes. More than 20% of the original user queries were taken as a sample for use in the control groups and tours were generated without paying attention to the link types described in section 3. The only criteria used for generating the control group tours were the individual nodes' similaritity to the query. The nodes in the control group tours were presented in decreasing order of node weight only. The results of this experiment can be seen in table 6 which compares connectivity scores between sequenced and unsequenced tours. Clearly, the overall scores are much lower for the control group than the assignments where sequencing of nodes was used.

μ value	User Sessions	Control Group
0.0 - 0.3	15%	63%
0.3 - 0.6	28%	21%
0.6 - 0.9	35%	7%
0.9 - 1.3	15%	3%
1.3 - 1.6	5%	2%
1.6 - 3.0	2%	4%

Table 5. Tour scores in user sessions v control

The figures in table 6 indicate that tours generated using the precedence ordering consistently obtain higher connectivity scores than those in the control groups suggesting that imposing a precedence order on a sequence of nodes improves their logical ordering.

By examining the patterns for each threshold value μ, it can be seen that the proportion of tours within each threshold value with connectivity scores above average generally increases as μ increases. This would indicate that the higher the value of μ the better. However, as μ increases the number of tours scoring below average also increases, e.g. 6% at $\mu = 0.1$ to 26% at $\mu = 0.6$. This indicates that as μ increases so does the number of low scoring tours. A suitable mid-point would seem to be where $\mu \approx 0.3$.

We cannot draw any definite conclusions just by examining users answers to the questions at the end of each session. However, we can gain some idea as to which type of tour is preferred by the users. From our users responses we see the tendency is towards short tours and then if necessary, deviate on their own. Our hypertext browser generated a recommended guided tour which users could follow if they chose and they could also deviate if they wished. In practice we found some of the user sessions (40%) contained no deviations from the recommended guided tour at all. However, on average, 7 deviations per session took place in the first assignment and 6 deviations per session in the second assignment. This suggests that users were using the hypertext browsing facility regularly in addition to following the guided tours.

6 Conclusions

Dynamically planning guided tours overcomes three problems with using hypertext in the following ways:

- *Getting lost:* Users don't need to worry about getting lost or confused in the hypertext as all they need to do is follow the tour which is generated in response to their query. If users want to digress and browse around the hypertext the option of returning to the tour at any time is always available.

- *Finding information:* When the user enters a search query in the form of a natural language statement it is matched to the nodes in the hypertext using standard information retrieval techniques. Unique hypertext characteristics such as connectivity and link types are also taken into account when determining which nodes should be included in the guided tour. By considering all characteristics of the hypertext we hope to incorporate the flexibility of browsing with the need to extract all relevant nodes.

- *Logical sequence of nodes:* When users are employing the hypertext as a learning aid it is important that there be a natural progression from the most basic information to the more complex so that the users do not become confused with the material. Our guided tours are planned so that the typed links between the nodes in the hypertext are considered in judiciously formulating the route. The most logical path through the network does not necessarily have to consist of nodes which are explicitly linked to one another but rather can be a combination of nodes selected from all parts of the hypertext. Selecting the nodes for inclusion is mainly an information retrieval task but planning the sequence of nodes depends on arranging the tour so that each node is positioned in the correct sequence relative to the other tour nodes.

User responses from nearly 1000 tours for 125 users have helped us refine our tour generation method. The present method generates a tour in response to a user's query irrespective of who the user is, their level of expertise or experience with the hypertext, and what he/she has seen already in the current or in previous sessions with the browser. The next step is to use this kind of information which is already stored in the log file to generate a user model and to use this to generate what we believe would be even more intelligently-planned dynamic guided tours. We are also considering ways of including relevance feedback during the tour traversal to dynamically re-compute a guided tour as has been done in [4]. We will also considerably expand the information coverage in the hypertext we use to allow us to involve an even larger population of end users.

References

[1] D. K. Barman, RelType: Relaxed Typing for Intelligent Hypermedia Representations, *Brown University Technical Report, CS-91-26,* April 1991.

[2] J.H. Coombs, Hypertext, Full Text, and Automatic Linking, *in: Proceedings of the 13th International Conference on Research and Development in Information Retrieval,* Brussels, Belgium, 83–98, 1990.

[3] M.D. Dunlop and C.J. van Rijsbergen, Hypermedia and Probabilistic Retrieval, *Proceedings of RIAO'91,* Barcelona, 337–356, 1991.

[4] M.E. Frisse and S.B. Cousins, Query by Browsing: An Alternative Hypertext Information Retrieval Method, *in: Proceedings of the 13th Annual Symposium on Computer Applications in Medical Care,* 388-391, Nov 1988.

[5] M.E. Frisse, Searching for Information in a Hypertext Medical Handbook, *Communications of the ACM, 31,* 881–886, 1988.

[6] R. Furuta and P.D. Stotts, Programmable Browsing Semantics in Trellis, *in: Proceedings of Hypertext '89,* Pittsburgh, 27- 42, 1989.

[7] S.H. Gray and D. Shasha, To Link or not to Link? Empirical Guidance for the Design of Nonlinear Text Systems, *Behavior Research Methods, Instruments & Computers, 21(2),* 326–333, 1989.

[8] N. Hammond and L. Allinson, The Travel Metaphor as Design Principle and Training Aid for Navigating around Complex Systems, *in: Proceedings of the 3rd Conference of the British Computer Society Human-Computer Interacion Specialist Group,* 75–90, 1987.

[9] C.C. Marshall and P.M. Irish, Guided Tours and On-Line Presentations: How Authors Make Existing Hypertext Intelligible for Readers, *in: Proceedings of Hypertext'89,* Pittsburgh, 1989.

[10] J Neilsen, Hypertext and Hypermedia, *Academic Press,* 1990.

[11] M.F. Porter, An Algorithm for Suffix Stripping, *PROGRAM, 14(3),* 130–137, 1980.

[12] G. Salton, Automatic Information Retrieval: The Transformation, Analysis and Retrieval of Information by Computer, *Addison- Wesley,* 1989.

[13] A. F. Smeaton, Prospects for Language-Based Information Retrieval, *Online Review, 15(6),* 373-382, 1992.

[14] M.A. Shepherd, C.R. Watters and Y. Cai, Transient Hypergraphs for Citation Networks, *Information Processing & Management, 26(3),* 395–412, 1990.

[15] R.H. Trigg, Guided Tours and Tabletops: Tools for Communicating in a Hypertext Environment, *ACM Transactions on Office Information Systems, 6(4),* 398–414, 1988.

[16] J.A. Waterworth and M.H. Chignell, A Model for Information Exploration, *Hypermedia, 3(1),* 35–58, 1991.

[17] P. Wright, Cognitive overheads and prostheses: some issues in evaluating hypertexts, *Proceedings of Hypertext '91,* San Antonio, 1991.

[18] P. Zellweger, Scripted Documents: A Hypermedia Path Mechanism, *Proceedings of Hypertext'89,* Pittsburgh, 1989.