BROWN UNIVERSITY
Department of Computer Science
Master's Thesis
CS-92-M10

''Browsing in Hypertext Documents With the Assistance of Automatic Link Generation''

by

Daniel Ta-Ping Chang

Browsing in Hypertext Documents With the Assistance of Automatic Link Generation

Daniel Ta-Ping Chang

Department of Computer Science
Brown University

Submitted in partial fulfullment of the requirements for the degree of Master of Science in the Department of Computer Science at Brown University

May, 1992

Date 5/14/92

Andries van Dam
Advisor

Browsing in Hypertext Documents With the Assistance of Automatic Link Generation

Daniel T. Chang
May 12, 1992

Graphics Group
Brown University
Providence, RI. 02912

ABSTRACT: The ability to create and browse collections of links in hypertext systems is considered their strongest advantage over traditional information retrieval (IR) systems [Conk88]. Most hypertext systems today facilitate link creation, but only a few support automatic link generation. In addition, current systems often require the user to supply a set of explicit keywords or tags; they give the user no control over the link generation process and often ignore user feedback [Coom90]. We have explored the enhancement of the user's browsing experience by the creation of personalized links that encapsulate user's personal interests, experience and knowledge. This paper describes our prototype system, HieNet, which records user-created links in a link profile in the database. By analyzing the link profile, the system creates a set of new links that may fit the user's browsing interest. In addition, the link creation process can be refined through several user-adjustable parameters.

KEYWORDS: Link, term weights, term vector, similarity measure, automatic link generation.

1. Introduction

Hypertext/hypermedia consists of nodes (or 'chunks') of information and links between them. A *link* in these systems connects one chunk of material, e.g. text (*source node*), to another chunk (*destination node*) to create a directed graph of nodes. Displaying and browsing potentially huge number of links often demands extensive cognitive processing on the user's part, and the user can easily become disoriented and "lost in hyperspace" (information space) [vanD88][Utti89]. Our intent is to reduce some of the problems associated with hypertext browsing/navigation. Some of the well-known strategies that have been implemented in our system to reduce the disorientation problems are described below.

(a) Extracting Hypertext Document Structures

Proponents of hypertext contend that nonlinear information processing mirrors two natural patterns of human information processing -- associative network structure and hierarchical structure. For example, the *book* metaphor represents hypertext documents using the hierarchical table-of-contents (TOC) structure, augmented by footnotes and cross-references

to provide associations. This technique is used to construct a mental map according to the user's preconceptions. The TOC provides landmarks while the user is browsing in the hypertext information space.

(b) Link Filtering

In order to reduce the sheer number of links, in particular irrelevant links, experts introduce the notion of typed links--links with attached attributes that can be used to group, sort and filter links [Enge68]. For example, the user can ask the system the equivalent of "Show me all the annotation links added to my term paper since last Friday by my English professor."

We found, however, that even after links are nicely organized and displayed, they still lack the personal touch. It's one thing for the user to traverse all the links displayed in the hypertext/ hypermedia system, it's another and much better thing for the system (or a personalized agent) to guess intelligently what links the user is most interested in seeing. We have focused our research on links between nodes that contain chunks of text. We want to capture the content representation of both the source node and destination node of a link by adding new attributes to the link table (see Section 2.2.1 and Figure 2). We call the augmented link table created in this way the *link profile*, and use it to analyze the user's interests and to create links automatically that mimic the user's linking pattern or browsing behavior.

With this goal in mind, we have constructed a HieNet system built on top of *Dyna*Text, a hypertext browser from Electronic Book Technologies, Inc. For a snapshot of the system, see Appendix B. Before we elaborate on HieNet's automatic link generation, we first introduce some fundamental *Dyna*Text features which HieNet utilizes.

2. Existing Features in *Dyna*Text

The *Dyna*Text graphical browser consists of a table of contents (TOC), a full-text window for displaying the text, and a Link Viewer with which the user can display and filter links. *Dyna*Text accepts documents that are tagged in the Standard Generalized MarkUp Language (SGML) by any standard SGML-Authoring Systems and formats them as an online hypertext. For example, one common way to mark up a document is to delineate its structure through a set of tags such as `<chapter>`, `<section>` and `<subsection>`. Tags can also be used to indicate typed links by specifying link attributes, and users can view desired links via filters. Another way to filter the documents is through the structure of the hypertext document itself. For example, the user can ask, "Show me all the video links (links to a piece of video) contained in the current section" or "Show me all the links in chapter 2." *Dyna*Text requires the document to be properly tagged in SGML before it can extract the hierarchical structures to form a tree. In our paper, we refer the tree as the *document tree* (see Figure 1). A collection of document trees forms a *document space.*

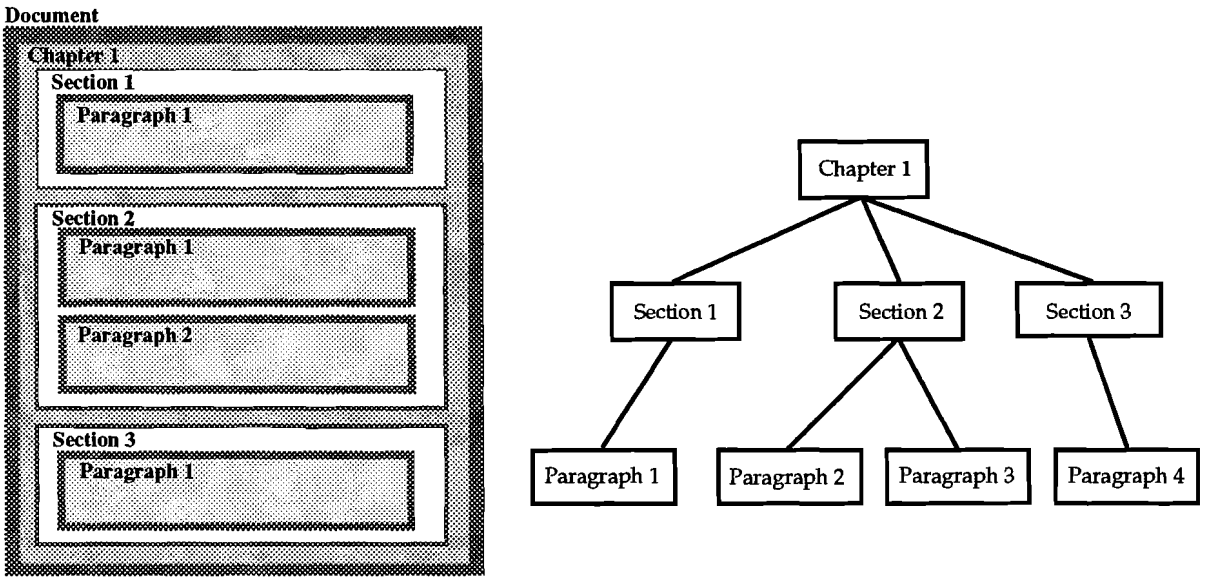


Figure 1. A document (left) and its corresponding document tree (right).

*Dyna*Text already supports a simple kind of "automatic link generation" that is accomplished through the tag attribute. For example, the tag `<video num=234>This is a text description of emergency procedure</video>` triggers the system automatically to create a link from this tagged text to video clip number 234 stored in the database. Another type of automatic link creation can be achieved through the built-in full-text searching mechanism of *Dyna*Text. For example, the system can create links between all the sections that resulted from user queries.

3. HieNet's Features

3.1 Vector Space Model

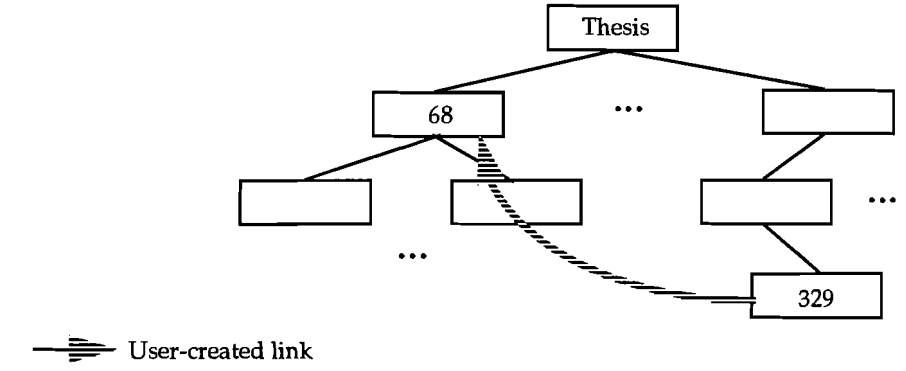
In current hypertext systems, links are typically stored in a database and each link is a record entry in the link table. Several fields/attributes are associated with a link, such as addresses for the source and destination node, link type, link owner and link date, but no information is stored in the link table that describes the content of the nodes.

HieNet adds extra fields to store the content representation of the link's source and destination nodes into the link profile. To accomplish this task, we have adopted the vector space model developed by the information retrieval (IR) community [Salt89]. Given a text node D_i, a content representation may be given as a *term vector* of terms $\overline{D}_i = (d_{i1}, d_{i2}, \ldots, d_{it})$, in which d_{ik} represents a *term weight*, of term T_k assigned to node D_i. The weight for each term d_{ik} is calculated using the

heuristic *term-weight equation* (term frequency divided by document frequency) that assigns high term weights to terms that occur frequently inside a particular node but relatively rarely in the document space (1).

$$d_{ik} = \frac{tf}{df} \qquad (1)$$

Note that $d_{ik} <= 1$, since the term frequency (number of times the term occur in a document) is at most equal to document frequency (total number of times the term occurred in the total document). Terms with a high term weight are known to be important in content identification [Salt89b]. Terms occurring with extremely high frequency in the document space turn out to have very small term weights because their document frequency is extremely high.



Type	Owner	Date	...	Source Addr	S. Vector	Destination Addr	D. Vector
text	dtc	3/31/92	...	Thesis.329	(0.34, 0.23, ..., 0.6)	Thesis.68	(0.03, 0.2, ..., 0.7)

Figure 2. At top, a user-created link in a document tree; below, the link profile in which the user-created link is recorded.

HieNet processes the document tree and composes a term vector for each of the nodes in the tree. In addition, whenever the user creates a link, HieNet uses the term-weight equation (1) to compose two term vectors, one for the source node and the other for the destination node, and stores them in the link profile (see Figure 2).

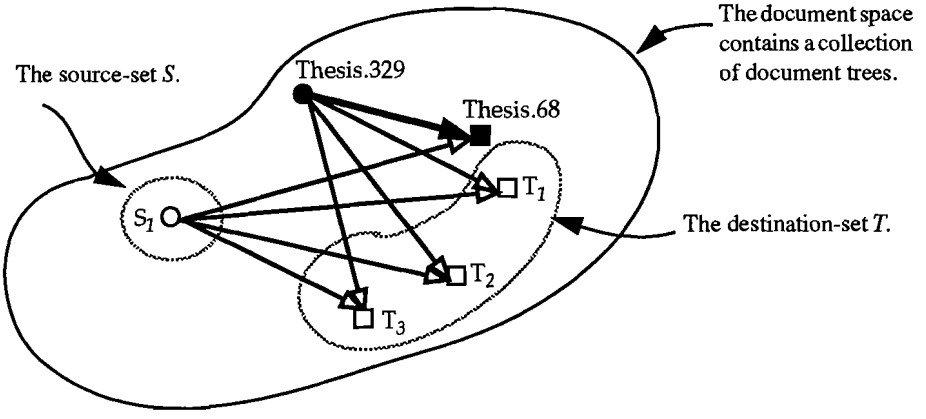
3.2 Similarity Measure Calculation

Given two nodes D_i and D_j, a *similarity measure* can be obtained between items based on the similarity between the corresponding term vectors. The similarity measure can be defined as an inner vector product (2) [Salt89][Salt89b]:

$$sim(D_i, D_j) = \sum_{k=1}^{t} d_{ik} \bullet d_{jk} \qquad (2)$$

3.3 Automatic Link Generation

Using the similarity measure equation, HieNet scans the document space to find node pairs whose similarity measures matches with the user-created links. First, any node whose similarity measure with the source vector in the link profile is above a certain threshold is put into the source-set *S*. Second, the same operation is applied to generate the destination-set *T*. Finally, HieNet creates links between the sets *S* and *T* (see Figure 3). An automatically created link is called a system-created link.



User-Created Link from Source Node (Thesis.329) to Dest. Node (Thesis.68)
System-Created Links
Source Node of the User-Created Link
Destination Node of the User-Created Link
Source Node of the System-Created Link
Destination Node of the System-Created Link

Figure 3. Conceptual view of link generation.

3.4 Relevance Ranking

Each destination node in the link is assigned a score calculated from the similarity equation (2). After the system completes the generation of the automatic link, a source node may contain more than one link to various destination nodes. The user is presented with a list of ordered destination nodes with the highest similarity measure displayed on the top of the list (see Figure 4).

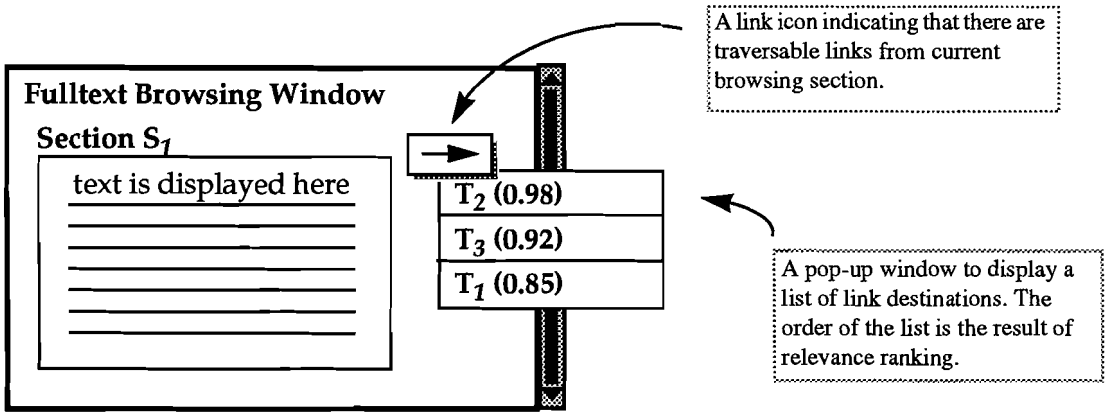


Figure 4. Snapshot of the fulltext browsing window.

3.5 User-Controlled Parameters

(a) Stoplist

The user can give HieNet a list of words to be eliminated at the start of the link-generation process. This process is useful in eliminating filler words such as "and", "or" and "not," or any other words that occur extremely frequently in the document space and are insignificant because their term weights are extremely small.

(b) Vector Term Selection

As a default, HieNet selects a group of terms whose term frequencies are closest to the median term frequency in the document space. Terms with extremely high or low occurrence frequency are not selected, since their term weights are close to zero and they are ineffective in the similarity measure calculation. Better term selection generates more relevant links for the user. This is one of the precision controls available in our system for link generation.

(c) Link Filters

The system does not create links indiscriminately. The user can apply filters to eliminate irrelevant links and use the remaining links to generate the link. For example, the user can filter out links created at certain date or by certain person, since all the links created in the system have time and creator attributes attached to them.

(d) Node Size

The user can adjust the size parameter to view links that contain nodes of a certain size. This is another precision control for link generation.

(e) Similarity Threshold

The user can adjust the similarity threshold to increase or decrease the number of links

created. Another interesting phenomenon is that when the similarity goes above a certain level, the system creates a link with larger node-size. The reason is that when nodes become larger nodes they also contain more words and their term frequencies are higher, which contributes to higher term weights and higher similarity threshold.

3.6 Scenario

The user can create a link profile for any given occasion. For example, the user can create a new link profile used in an art class and save the link profile for other art-related hypertext documents. The user can also load other user's link profile to generate new links.

4. Implementation

HieNet consists of approximately 2,000 lines of code in C programming language. The conciseness of the code can be credited to the powerful Application Programer's Interface of the *Dyna*Text's Toolkit and its clever software design.

The following sections describe the main modules of the HieNet system.

4.1 Preprocessing

(a) Term Elimination

All terms in the document space specified in the stoplist are excluded from the link generation process. The user can modify the stoplist or supply a different one.

(b) Term Frequency Calculation

The term frequencies of the remaining terms are found (term frequency is simply the number of times a term occurred in the document space).

(c) Vector Term Selection

A default number of terms whose occurrence frequencies are closest to the median term frequency are selected for use in the term vector. The user can also select different terms.

(d) Vector Term Weight Calculation (Pre-Order Traversal of the Document Tree)

Using pre-order traversal of the document tree, a term vector is calculated for each node in the tree. This is implemented with a recursive function; the time complexity is $O(n)$, where n is the number of nodes in the document tree.

(e) Size Calculation (Post-Order Traversal of the Document Tree)

In the tree-traversal process (in step d), we also keep track of the size of the current section (node), which is the sum of all its children's sizes. When a section has no child, then the size is

simply the number of words in the section. This is also implemented with a recursive function. The size calculation is done in post-order traversal of the document tree, and the time complexity for this function is also O(n).

4.2 Storing the User-Created Link in the Link Profile

Every time the user creates a link, a link record is added to the link profile. This link record includes the creator, creation time and link type attributes and, most importantly, a pair of term vectors, one for the source node and one for the destination node.

4.3 Automatic Link Generation Based on the Link Profile

A link is created on a node in the document tree if and only if none of its children have links created on them. HieNet uses post-order tree traversal to create a link on a node that is as small as possible. Only when this attempt fails does the system create a link on a larger section. We use this strategy because we believe the user is most likely to be interested in reading links that connect sections containing the fewest words, such as paragraphs. In situations where the user wants to see links on sections with larger chunks of text, s/he can simply increase the node size parameter with a slider provided by the system (see Appendix A). This recursive function has a upper bound of O(n). In practice, the time complexity is drastically lower than O(n), because HieNet checks if the current node has a similarity measure above the threshold; only if so does it traverse the descendents of the node; otherwise it ignores the subtree of the node.

5. Comparison of Our Model with Related Work

Salton started using the similarity equation (our Equation 1) in his automatic linking research [Salt89b]. In his model, any two nodes that have similarity above a certain threshold are connected together to form a link. We found the main disadvantage of his method is that the links are always identical for everyone who uses the system, since the user has no control over link generation.

By contrast, our automatic link generation is based on user-created links. Based on the link profile, HieNet finds new sets of source nodes and of destination nodes and generates a link between the two sets. In addition, in Salton's model, links were created on a fixed-size nodes, i.e., from paragraph to paragraph, while in our model, there are no constraints on node size. Links can be generated from nodes of any size node to nodes of any other size. For example, even if the user creates a link from paragraph to paragraph, the system would create links either from a paragraph to a paragraph or from a section to a paragraph, depending on whether the node satisfies the similarity threshold. The user can browse nodes with a dynamic range of sizes. This is the most "intelligent" behavior exhibited by our system, in that it is able to tell whether the whole (the chapter node) is more relevant than its parts (the paragraphs) or vice versa.

6. Experimental Hypertext Link Generation

Seven articles related to the recent Los Angeles riots were transcribed from *Newsweek* and *Time* into SGML-based HieNet documents. Two chapters of an object-oriented graphics package manual were intermingled with the riot articles in the document space. These sample corpuses consist of about 16,000 words. Table 1 contains a sample set of terms selected by the system to be used in the term vector.

The first set of experiments were rather crude, involving constructing two types of links, one that links nodes in the graphics package sections and the other that links nodes in riot sections. HieNet was able to create new links that matched the original link's content, i.e., it did not create any links between the graphics package section and riot sections.

The second set of experiments was focused on whether HieNet created any useful or meaning links. The results are shown in Table 2. While judging how well HieNet performed is somewhat subjective, we found that it produced many meaningful links but at the same time also produced too many links that the user found irrelevant. For example, to see links that connected nodes with paragraphs, not sections, the user had to struggle with the various controllers.

We realized that, while providing the user with many controllers for adjusting parameters is a good thing, it could be very confusing. Table 3 shows the dramatic effect of changing the threshold controller for the source node. To reduce this problem we would like to automate the size parameter. We can record the user-link's node size into the link profile so as to generate links with similar node sizes.

We also detected that some very relevant links were missed because the term selection was relatively inadequate and incomplete for some sections and thus the system was unable to create links for those sections. A naive way to correct this, at a high cost in space and speed, is to increase the number of terms used by the term vector. We also would like to find a better term-selection algorithm in the immediate future.

7. Future Research

Currently, HieNet has only been tested on a small set of documents. In the future we plan to test HieNet on users with large-scale documents in order to verify and fine-tune the various heuristics and assumptions used in our research. In addition, we considered some possible extensions to increase HieNet's effectiveness in automatic link generation:

(a) Link Generation via Natural Language Understanding

Ultimately, we would like to use a natural language understanding (NLU) engine that can "interpret" the meaning of words. The NLU engine should dramatically increase the precision of link generation.

(b) Link Version Control

The result of automatic link generation could vary if the user uses different link profiles or as more hypertext documents are added into the document space. The user may want to keep a history (sequence of snapshots) of the system-created links, so as to be able to review how various links evolved in the document space.

(c) Automatic Group-Link Generation via Group-Link Profile

The user can ask the system to create links that are relevant to the group/corporate interests by using group/corporate link profiles. We can simply add this feature by introducing a new field into the link profile that can be used to distinguish types of groups. We can then use filters to extract group-related links to generate links.

(d) Incorporate Link Location into the Similarity Measure Equation

Our current model does not consider a link's document position in the weighting equation. It seems reasonable to assume that a link originating at a chapter title should have higher term weight than a link originating from a paragraph. Quantifying useful weighting parameters is a challenge for the future.

8. Conclusion

Preliminary tests show that HieNet is able to assist the user of our hypertext system by displaying links that are personalized toward the user's browsing interest. We have described a model that integrates practical IR techniques into SGML-based hypertext document systems. Secondly, HieNet introduces sound data structures for storing link-content and efficient algorithms for the automatic link generation. Finally, HieNet parameterizes many useful heuristics and gives the user direct access to and manipulation of those parameters.

9. Acknowledgments

I wish to express my gratitude to Steve DeRose, Jeff Vogel and Dave Sklar, principal architects of *Dyna*Text from Electronic Book Technologies, for providing the latest version of their software and for their helpful and stimulating commentary; to Paul Kahn, senior researcher at Brown University's Institute for Research in Information and Scholarship, for his insightful criticism of this paper; and to my advisor Professor Andries van Dam, for his relentless inspiration.

10. References

[Bern90] Mark Bernstein, "An Apprentice That Discovers Hypertext Links", *Hypertext: Concepts, Systems and Applications, Proceedings of the European Conference on Hypertext*, Versailles, France, November 1990.

[Bush45] Vannevar Bush, "As We May Think", *Atlantic Monthly*, July 1945.

[Conk88] Jeff Conklin, "Hypertext: A Survey and Introduction", *IEEE Computer*, vol. 20, no. 9, September 1988.

[Coom90] James Coomb, "Hypertext, Full Text, and Automatic Linking", *Proceedings of SIGIR '90 ACM*, New York, 1990.

[Deyo90] Laura De Young, "Linking Considered Harmful", *Hypertext: Concepts, Systems and Applications, Proceedings of the European Conference on Hypertext*, Versailles, France, November 1990.

[Enge68] Douglas Engelbart and William English, "A Research Center for Augmenting Human Intellect", *Proceedings AFIPS Conference*, 1968 Joint Computer Conference. December 9-11, 1968, SF. Montvale, NJ: AFIPS Press, 1968, pp. 395-410.

[Fein81] Steven Feiner, Sandor Nagy, and Andries van Dam, "An Experimental System for Creating and Presenting Interactive Graphical Documents", *ACM Transactions on Graphics* 1(1), pp. 59-77, 1981.

[Fein88] Steven Feiner, "Seeing the Forest for the Trees: Hierarchical Display of Hypertext Structure", *ACM Conference on Office Information Systems*, Palo Alto, CA, March 23-25, 1988.

[Furn86] George Furnas, "Generalized Fish-Eye Views", in *Proceedings of the 1986 ACM Conference of Human Factors in Computing Systems*, CHI '86, April 1986.

[Robe91] George Robertson, Jock Mackinlay, and Stuart Card, "Cone Trees: Animated 3D Visualizations of Hierarchical Information", *Proceedings of the ACM SIGCHI '91 Conference on Human Factors in Computing Systems*, pp. 189-194, April 1991.

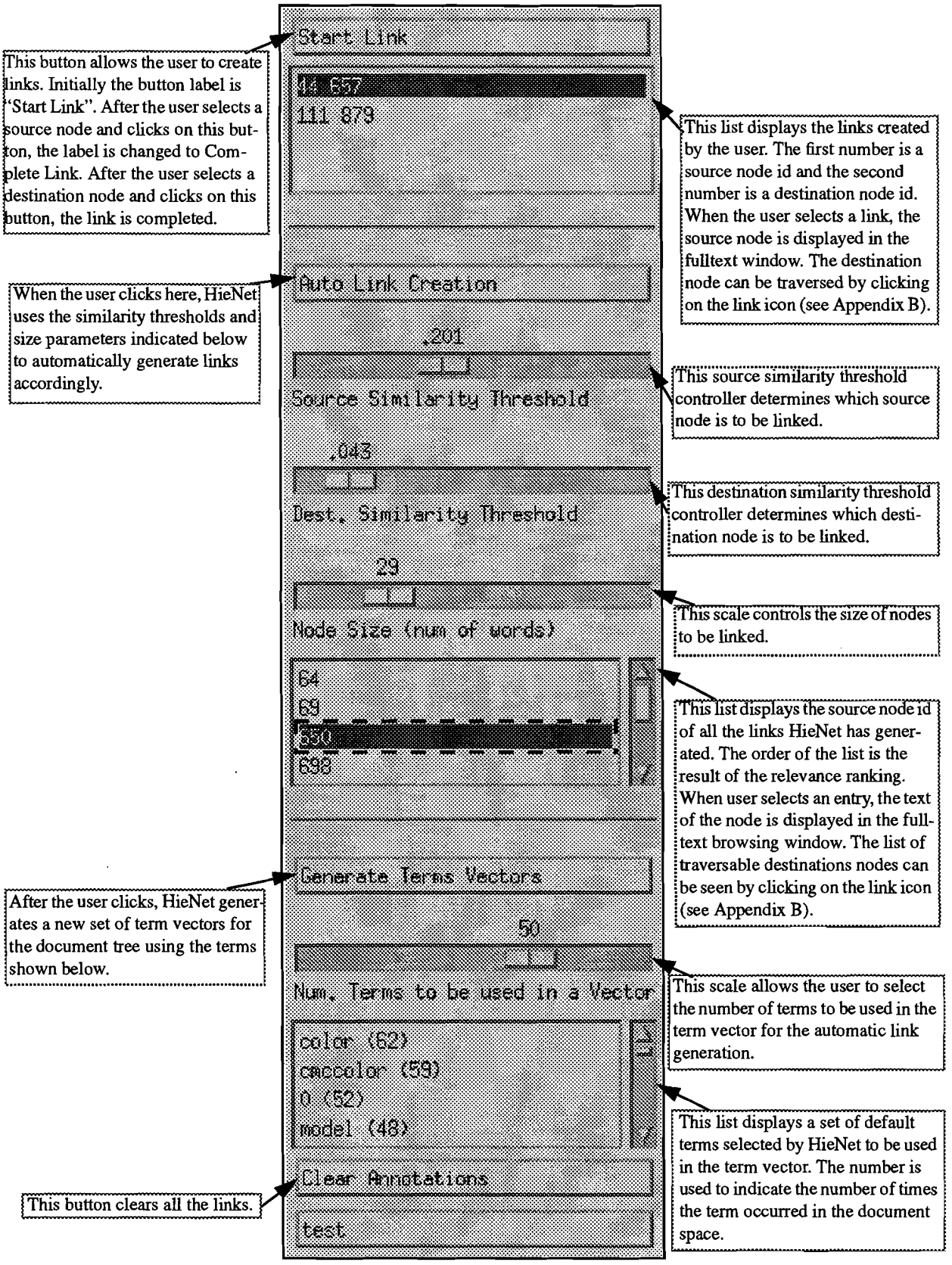
[Salt89] Gerald Salton, *Automatic Text Processing*, Addison-Wesley, Reading, MA, 1989.

[Salt89b] Gerald Salton, "On the Automatic Generation of Context Links in Hypertext", TR 89-993, Cornell University, Ithaca, NY, April 1989.

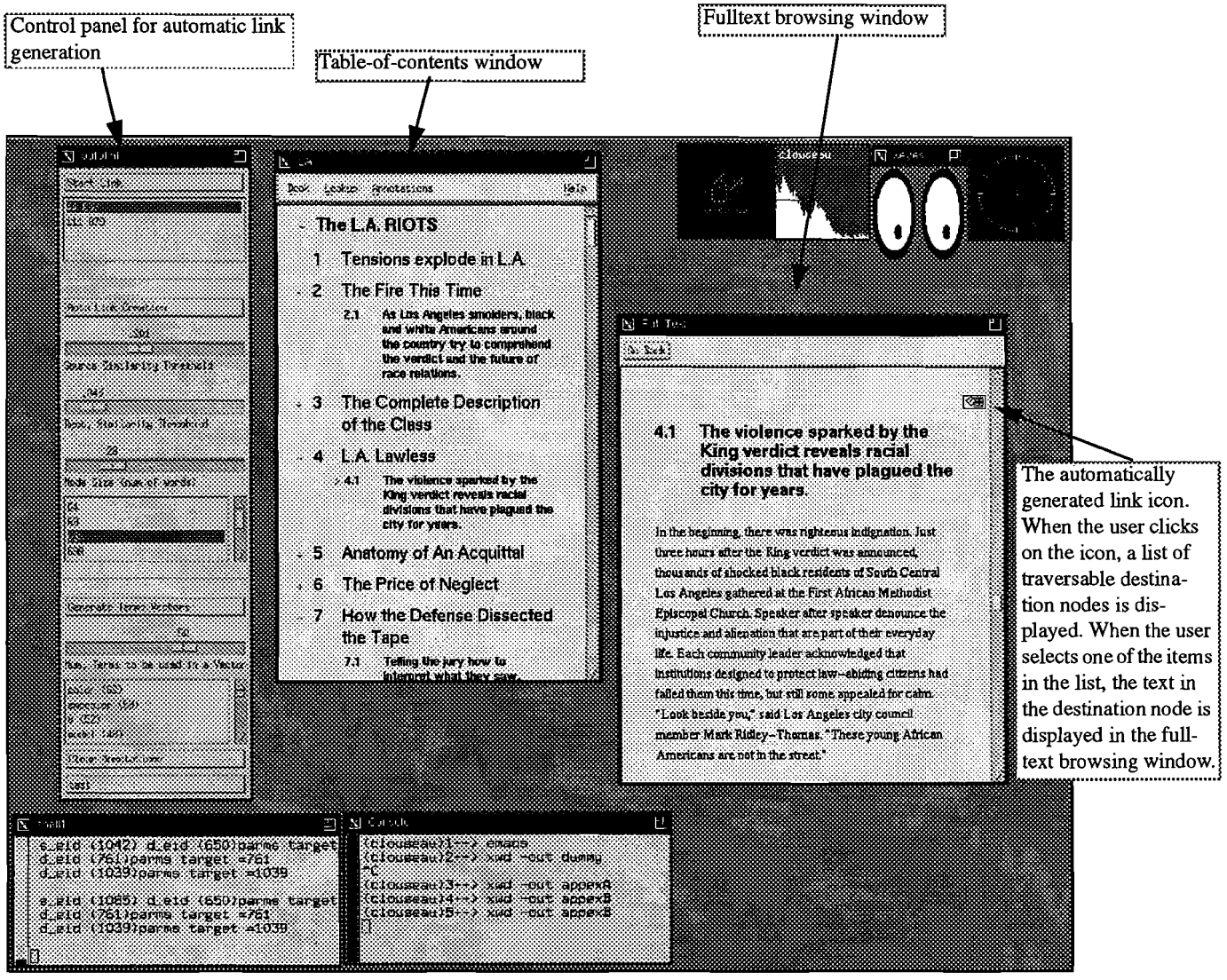
[Utti89] Kenneth Utting and Nicole Yankelovich, "Context and Orientation in Hypermedia Networks", *ACM Transaction on Information Systems*, vol. 7, no. 1, p. 58-84, January 1989.

[vanD88] Andries van Dam, "Hypertext '87 Keynote Address", *Communications of the ACM*, vol. 31, no. 7, July 1988.

[Yank85] Nicole Yankelovich, Norman Meyrowitz, and Andries van Dam, "Reading and Writing the Electronic Book", *IEEE Computer*, 1985.



Appendix A. Control panel for automatic link generation. The top list displays user-created links, the middle list displays the system-created links, and the bottom list shows terms used in the term vector.



Appendix B. Snapshot of the HieNet System. The top left window is the control panel for automatic link generation, the middle window is the Dyna*Text* TOC and the rightmost window is the full text browsing win-

Terms	Document Frequency
color	62
model	48
blacks	47
values	37
jury	37
defense	37
class	32
beating	32
verdict	30
law	24
force	24
african	24
cops	20
riots	19
violence	17
simi	17
video	16
rodney	14
trial	13
poverty	12

Table 1:
Sample terms selected by HieNet for the term vector.

	Source Node (S Thresh. 0.09)	Destination Node (S Thresh. 0.12)
User-Created Link	Less than three ***hours*** after acquittal verdicts were read for four Los Angeles police officers, gangs of angry, young black ***men*** took to the ***streets*** in search of their own brand of justice.	But it was already too late. By the time the two-hour meeting broke up, the first fires had been set. As weary parishioners left the prayer meeting, some were ***shot*** at by ***rioting*** thugs. "Nothing you're ***talking*** about is going to do any good," one young man told the departing crowd. "Come with us--let's ***burn***."
System-Created Link	Other law-enforcement units were also slow to react. Though California Governor Pete Wilson deployed about 2,000 National Guard troops on Wednesday evening, it took almost 24 ***hours*** for the extra ***men*** to reaching the ***streets***...	Worse, the ***riots*** demonstrated again the existence of a group of mostly young, impoverished and angry ghetto blacks who no longer listen to the established African-American leadership--or to anybody..."Nothing you're ***talking*** about is going to do any good--so come with us and let's ***burn***." Some rioters even ***shot*** at the churchgoers...
System-Created Link	For more than 48 ***hours***, an urban nightmare came true as hatred ruled the ***streets***. During that time, parts of the city virtually ceased to function...	For the black majority, fear and fury do not translate into approval of--let alone participation in--***rioting***. Apart from moral considerations, blacks realize that it is their neighborhoods that ***burn*** and mostly their lives that are lost.
Terms Contributing High Term Weight in the Term Vector	hours, men, streets.	shot, burn, talk(ing), riots, rioting.

Table 2:

An example of automatic link generation. The first row is the user-created link. The remaining rows (2-4) are the system-created links (link). The last row indicates the terms that contributed to the similarity measure and fired the links.

Source Node Threshold Value	Source Set S	Node Tag Name
0.2	3	document
0.077	761	chapter
0.04	44	section
	698	section
	761	chapter
	823	section
	1039	chapter
0.009	6	chapter
	49	paragraph
	64	paragraph
	650	section
	701	paragraph
	703	paragraph
	769	paragraph
	785	paragraph
	791	paragraph
	807	paragraph
	841	section
	1053	paragraph
	1073	section
	1085	paragraph

Table 3:

The effect of decreasing the source-node threshold effects on HieNet's selection of nodes into the source set S.