

Silk from a Sow's Ear: Extracting Usable Structures from the Web

Peter Pirolli, James Pitkow, Ramana Rao

Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304, USA

E-mail: {pirolli, rao}@parc.xerox.com, pitkow@cc.gatech.edu

ABSTRACT

In its current implementation, the World-Wide Web lacks much of the explicit structure and strong typing found in many closed hypertext systems. While this property has directly fueled the explosive acceptance of the Web, it further complicates the already difficult problem of identifying usable structures and aggregates in large hypertext collections. These reduced structures, or localities, form the basis to simplifying visualizations of and navigation through complex hypertext systems. Much of the previous research into identifying aggregates utilize graph theoretic algorithms based upon structural topology, i.e., the linkages between items. Other research has focused on content analysis to form document collections. This paper presents our exploration into techniques that harness both the topology and textual similarity between items as well as integrate new analyses based upon actual usage of the Xerox's WWW space. Linear equations and spreading activation models are employed to arrange Web pages based upon functional categories, node types, and relevancy.

Keywords

Information Visualization, World Wide Web, Hypertext.

INTRODUCTION

The apparent ease with which users can click from page to page on the World-Wide Web (WWW) belies the real difficulty of understanding the what and where of available information. The primary approaches widely provided for finding information are search systems like Lycos and Harvest and human-organized directory browsers like Yahoo and Internet Yellow Pages [8]. Though these approaches are quite powerful, they don't exhaust the potential space of approaches or needs. For example, they provide very little support for helping people assimilate the structure and content of Web

localities.

An alternative approach involves providing an interactive overview of such localities which facilitates navigation and general assessment. Information Visualization offers an approach to providing such an overview. Visualizations have been developed which provide new interactive mechanisms for making sense of information sets with thousands of objects [7]. The general approach is to map properties and relations of large collections of objects onto visual, interactive structures.

To the extent that the properties that help users navigate around the space and remember locations or ones that support the unit tasks of the user's work, the visualizations provide value to the user. Visualizations can be applied to the Web by treating the pages of the Web as objects with properties. Mukerjea [5] has utilized a number of visualizations on the Web. Each of these visualizations provide an overview of a locality in the Web in terms of some simple property of the pages. For example, a Cone Tree shows the connectivity structure between pages and a Perspective Wall shows time-indexed accesses of the pages. Thus, these visualizations are based on one or a few characteristics of the pages.

In this paper, we describe our approach to extracting structure in the Web which can be used to form higher level abstractions that reduce the complexity and increase the richness of an overview. We have developed methods for annotating pages with their functional types and relevancy/importance assessments as well as aggregating the Web into collections which can be treated as collections. Thus with this higher level structure, an overview can be outfitted with landmarks and aggregate objects which increase the richness while at the same time reducing complexity. Many kinds of information can be used to classify or organize collections of WWW pages including the textual content, the connectivity structure, and various characteristics of the pages including file-system like attributes and access statistics. Most of this information can be straight-forwardly gathered for fixed collections of pages, particularly with privileged access to the server's storage system. Over time, the Web

Table 1. Node type definitions and results.

Node Type	Size	Number Inlinks	Number Outlinks	Depth of Children	Similarity to Children	Freq.	Entry Point	Precision
Index	- (outlinks /size)		+					0.67
Source Index	- (outlinks /size)		+				+	0.53
Reference	+	-	-	-				0.64
Destination Reference	+	-	-	-			-	0.53
Head			+	+	+		+	0.70
Organization Home Page		+	+		+		+	0.30
Personal Home Page	> 1000 k < 3000 k					-	-	0.51
Content	+	-	-					0.99

infrastructure can be adapted to provide this information directly through standard protocols.

We have gathered information for several Web localities, but for the purposed of this paper, will focus on all the pages served by the Xerox WWW server. Our goal was to analyze this data and to design algorithms for various extractors which would annotate and aggregate the Web locality. In particular, we have designed methods for classifying nodes into a number of functional categories, spreading relevance based on selecting one or more source nodes and dimensions of interest, and aggregating nodes into higher level collections.

OVERVIEW OF THE APPROACH

We may conceive of a Web locality as a complex abstract space in which we have arranged Web pages of different functional categories or types. An example category might be *organizational home page*. Typical members of the category would describe an organization and have links to many other Web pages providing relevant information about an organization. Table 1, which is described below in detail, lists the functional categories that we attempted to extract in this study. Each functional category is defined in a manner that has a graded membership, with some pages being more typical of a category than others, and Web pages may belong to many categories.

The degree of relevance of Web pages to one another can be conceived as similarities among Web pages located in the abstract space. We will represent the similarity, or strength of association, relations using a composite of

several graph structures. Each graph structure contains nodes representing Web pages, and directed arcs among nodes are labeled with values representing strength of association among pages. One type of graph structure represents the link topology of a Web locality by using arcs labeled with unit strengths to connect one graph node to another when there exists a hypertext link between the corresponding Web pages. This is perhaps the most common intuition that people hold when thinking about a locality. A second type of graph structure represents the inter-page text content similarity by labeling arcs connecting nodes with the computed text similarities between corresponding Web pages. This is a common way of conceptualizing documents in search-based information retrieval. A third type of graph structure represents the flow of users through the locality by labeling the arcs between two nodes with the number of users that go from one page to another. This essentially reflects the mutual desirability of information among Web pages based on observed usage patterns.

Suppose a user is interested in a set of one or more Web pages and wants to find related pages to form a small Web aggregate. We assume that the identification of such "interesting seed sets" of Web pages could also be automatically determined based on functional type identification. Additionally, we use a spreading activation mechanism [1] to compute a degree-of-relevance function over a Web locality from a given set of source Web pages.

Conceptually, activation is pumped into one or more of the graph networks at nodes representing the starting set of Web pages and flows through the arcs defined above, with

the amount of flow modulated by the arc strengths (which might also be thought of as arc flow capacities). The asymptotic pattern of activation over nodes will define the degree of relevance of Web pages to the source pages. By selecting the topmost active nodes or those above some set criterion value, we may extract relevant aggregates of Web pages.

In summary, given this conception of a Web locality, we use three component processes to identify and extract Web structure. First is the categorization of Web pages into types and the users' selection of pages according to their degree of category membership in different types. Second is the spread of activation from the identified sources through some combination of the link connectivity, text similarity, and usage spaces. Third is the selection and aggregation of Web pages based on their pattern of relevancy as measured by activation.

DATA SOURCES AND COLLATION

The data used for subsequent analyses was derived from two sources: a traversal of the Xerox's external Web site <URL:http://www.xerox.com> and the logs of requested items maintained by the Xerox Web server. For this analysis, we choose the access logs from March through May of 1995, in which 1.4 million items were requested.

The site's topology was ascertained via "the walker", an autonomous agent that, given a starting point, performs an exhaustive breadth-first traversal of pages within the locality. The walker utilizes the Hypertext Transfer Protocol (HTTP) to request and retrieve items, parsing the returned object to extract hyperlinks. Only links that point to objects within the site are added to a list of items to explore. Thus, the walker produces a graph representation of the site, with each node having at least the following meta-information/properties: name, title, list of children, size, and the time the node was last modified. It is salient to note that the walker may not reach all nodes that are accessible via a particular server -- only those nodes that are reachable from the starting point and its children are included. This analysis produces an adjacency matrix for the particular locality.

Most servers have the ability to record transactional information about requested items. This information usually consists of at least the time and the name of the URL being requested as well as the machine name making the request. The latter field may represent only one user making requests from their local machine or it could represent a number of users whose requests are being issued through one machine, as is the case with firewalls and proxies. This makes differentiating the paths traversed by individual users from these access logs non-trivial, since numerous requests from proxied and firewalled domains can occur simultaneously. That is, if 200 users from behind an America Online proxy are simultaneously navigating the pages within a site, how does one determine which users took which paths? This

problem is further complicated by local caches maintained by each browser and intentional reloading of pages by the user.

The algorithm we implemented to determine user's paths, a.k.a. "the whittler", utilizes the locality's topology along with several heuristics. The topology is consulted to determine legitimate traversals while the heuristics are used to disambiguate paths when multiple users from the same machine name are suspected. The latter scenario relies upon a least recently used bin packing strategy and session length time-outs as determined empirically from end-user navigation patterns [3]. Essentially, new paths are created for a machine name when the time between the last request and the current request is greater than the session boundary limit, i.e., the session has timed out. New paths are also created when the requested page is not connected to the last page in the currently maintained path. These tests are performed on all paths being maintained for that machine name, with the ordering of tests being the paths least recently extended. This produces a set of paths requested by each machine and the times for each request. From this, a vector that contains each node's frequency of requests and a matrix containing the number of traversals from one page to another are computed using software that identifies the frequency of k-substrings for any n-string [6]. These are referred to hereafter as the frequency vector and the path matrix respectively.

Additionally, the difference between the total number of requests for a page and the sum of the paths to the page was computed. Intuitively this generates a set of "entry point" candidates. Entry points are defined as the set of pages that are pointed to by sources outside the locality, e.g., an organization's home page, a popular news article, etc. Table 2 shows the results of this analysis. As one would expect, the Xerox Home page is the most frequently requested entry point. Additionally, we see the PARC's Digital Library Home Page, PARC's Map Viewer, which is winner of two "Best of the Web 94" awards, the Bookwise Home Page (the world's first PC-based interactive system for teachers and their students) and the 1995 Xerox Fact Book are the top pages identified.¹ Besides providing useful insight to Web designers based on actual use, which may differ from their intended use of the site, the identification of entry points provide the set of nodes from which to spread activation.

Another source of information about relationship between pages is the similarity of their textual content. Techniques from information retrieval [9] can be straightforwardly applied to calculate a similarity matrix which provides a usable measure of this variable. In particular, for each page, we tokenized and indexed the text using the TDB full-text content engine which provides access to vectors

¹Footnote: an outlier was removed that resulted from a parsing error in the walker.

for each document in the space formed by the dimension for each term in the collection. For each pair of pages, we computed the dot product of these vectors to as a similarity measure in this space.

Table 2. The most popular starting points

% Visits Outside	Number Visits	Pages
99.96	2662	/95FactBook/Title.html
96.18	12377	/PARC/docs/mapviewer-legend-world.html
99.58	16004	/Products/XIS/BookWise.html
99.99	19130	/PARC/dlhx/library.html
94.29	24107	/ (the default Xerox Home Page)

LINK TOPOLOGY-BASED APPROACHES

Botafoga et al [2] have reported on purely graph theoretic techniques for splitting a hypertext into aggregates. These techniques are based on identifying articulation points in the undirected graph and removing them to create a set of subgraphs. A node is an articulation point if removing it and its edges would disconnect the graph. Botafoga et al describe two algorithms which repeat this procedure iteratively. These algorithms removes indices (nodes with relatively high number of out-links) and references (nodes with relatively lots of in-links) on each iteration. The logic of this is to prevent these functional nodes from overconnecting the graph. However, in our case, many of the nodes identified were in fact table-of-contents-like

nodes which are very important elements of a webgroup.

Applying their first algorithm to the graph structure of the Xerox Web produces 10 webgroups with at least 10 nodes, shown in Table 3. In addition, we tried a more simple algorithm which iteratively removes articulation points until all groups are below 25 nodes in size or contain no articulation points. In particular, we didn't remove indices or references during iteration. This leads to 9 clusters (again of at least 10 nodes), shown in Table 3. The two algorithms produced 8 webgroups in common, though often not including the same nodes. In addition, the simplified algorithm produced one extra webgroup, while the 2 extras webgroups produced by the Botafoga algorithm were caused by splitting a webgroup and by including a spurious webgroup.

Unsurprisingly, these algorithms were quite effective at pulling out very typically highly-connected book structures. For example, the 13 node "html_training/index.html" book was a TOC with 12 nodes for sections which pointed back and forth. These are essentially highly-authored sections of the web and cluster together in a number of ways. For example, again as would be expected, there was a very high correlation between the URLs of the nodes within these webgroup. Most of the nodes typically sharing a prefix of two or three pathname parts, though webgroups that were less book-like tended to also bring in a few nodes from other locations on the server.

Table 3. Web aggregation using link topology.

Cluster	No.	Node ID	First URL
Butafogo Method			
1	32	24	/liveworks/lwi_web/about.html 6
2	23	60	/XSoft/vrpeform.HTML 2
3	11	75	/Products/MajestiK/MajestiKSeries.html 3
4	20	97	/XPS/prodpage/4050.htm 0
5	14	333	/show/PressReleases/Overview.html 4
6	40	456	/PARC/spl/eca/oi-project.html 6
7	50	497	/digitrad/short 0
8	13	508	/PARC/people/jyu/html_training/index.html 0
9	22	615	/RXRC/Cambridge/trs/ps/1995/EPC-1995-101.ps 4
10	14	632	/RXRC/Cambridge/trs/ps/1994/EPC-1994-106.ps 2
Simplified Butafogo Method			
1	22	1	/printsolutions.html 1
2	35	9	/ic1.html 2
3	20	24	/liveworks/lwi_web/about.html 0
4	59	215	/digitrad 0
5	14	333	/show/PressReleases/Overview.html 4
6	32	456	/PARC/spl/eca/oi-project.html 2
7	13	508	/PARC/people/jyu/html_training/index.html 0
8	47	513	/RXRC/Cambridge/people/thumbnails.html 4
9	63	750	/PARC/spl/eca/oi/gregor-invite/P000.html 0

NODE TYPING

Previous hypertext research extols the value added of strongly typed node and link systems, yet most of the information available on the Web is poorly typed. Even so, a quick tour of pages across localities reveals that certain classes of documents do indeed exist. This next section presents the rationale, categories, and algorithms used to identify certain classes of Web pages, followed by a discussion of the results obtained from the Xerox Web locality.

As a first pass, we define the following types of Web pages:

- *head*: If one were to take a set of related pages and place one page as the first page people would visit this would be it (strengthen). Head pages have two subclasses:
 - *organizational home page*: These are pages that represent the entry point for organizations and institutions, usually found as the default home page for servers, e.g., <http://www.org/>
 - *personal home page*: Usually, individuals have only one page within an organization that they place personal information and other tidbits on.
- *index*: These are pages that server to navigate users to a number of other pages that may or may not be related. Typical pages in this category have the words "Index" or "Table of Contents" or "toc" as part of their URL.
- *source index*: These pages are that are also head nodes, those that are used as entry points and indices into a related information space.
- *reference*: A page that is used to repeatedly explain a concept or contains actual references. References also have a special subclasses:
 - *destination*: In graph theory these are best thought of as "sinks", pages that do not point elsewhere but that a number of other pages point to. Examples include pages of expanded acronyms, copyright notices, and bibliographic references.
- *content*: These are pages whose purpose is not to facilitate navigation, but to deliver information.

The solution path we took to determine the set of pages that comprise each class folds in the above usage, textual similarity, and meta-information for each item in the Xerox Web space. Specifically, a new matrix was created with each row representing an item and the columns representing the item's:

- *size*, in bytes, of the item
- *inlinks*, the number of hyperlinks that point to the item from the Xerox Web space

- *outlinks*, the number of hyperlinks the item contains that point to other items in the Xerox Web space
- *frequency*, the number of times the item was requested in the sample period
- *sources*, number of times the item was identified as the source node of a path traversal (improve definition)
- *csim*, the textual similarity of the item to it's children based upon previous tdb calculation
- *cdepth*, the average depth of the item's children, i.e., ((the sum of the depth of the item's children) - (the depth of the site)) / (the number of children), where depth is measured as the number of '/' in the URL

A logarithmic transform was applied to the size, inlinks, outlinks, frequency, & sources values after having a constant of one added to remove zero values. Additionally, z-scores were computed on the transformed data, resulting in a normalized columns in the matrix. Two additional matrices were derived from the original dataset, one with zero size items removed and the other with only item whose sizes were between 1000 and 3000 bytes.

Given the above properties and shapes of the distributions, linear separable categories were assumed. This enabled categories to be identified by solving a set of linear equations of the form:

$$c_i = w_1 v_1 + w_2 v_2 + \dots + w_n v_n \quad (1)$$

for all nodes i in Xerox Web space, where the v_j are the measured features of each Web page, and the w_j are weights.

For example, we hypothesized that Content Nodes would have few in and outlinks, but have somewhat larger sizes. The equation used to determine this category of nodes had a positive size weight and negative in and outlink weights. For Head Nodes, we reasoned that the contextual similarity between itself and its children would be high, that it would have a large number of children that would either be in the same directory or lower (positive depth), and that it would be more likely to be the source node based upon actual user navigation patterns. Table 1 shows the weights used in an ordering of the nodes for each category.

Once the set of nodes for each type were identified, the top 25 members with the highest solution of the linear equation were extracted and the first page of the corresponding Web pages printed for off-line evaluation. For each node type, each page was given either a 'belongs' or 'does not belong' ranking by three examiners (us) to determine precision, where precision is defined as the number of correctly included pages in the set. Table 1 also shows the average precision (geometric mean) for each node type.

Table 4. The top 5 head nodes.

URLs of Page	Titles of Page
1. /PARC/DigiTrad/DigiTradKeywords.html	Digital Tradition Keywords
2. /RXRC/Cambridge/trs/html/index.html	RXRC Cambridge Technical Report Series
3. /PARC/istl/gir/fishkin.html	Ken Fishkin's Public Home Page
4. /PARC/spl/eca/oi/gregor-invite/transcript.html	Why are Black Boxes so Hard to Reuse?
5. /Investor/10K-94-Part-IV-g.html	Xerox Corporation 1994 Form 10-K

As one would expect due the large number of content nodes in a Web locality, the precision at which content nodes can be identified is quite high (0.99). Equally encouraging is the identification of Head and Index Nodes (0.70 and 0.67 respectively). Table 4 shows the list of top five Head Nodes. Not surprisingly, the lowest precision in Table 1 is associated with the correct identification of Organizational Home Pages, of which only less than ten pages within the Xerox Web space belong.

SPREADING ACTIVATION

We use spreading activation as a means for identifying localized subareas of a Web locality. Spreading activation can be characterized as a process that identifies knowledge relevant to some focus of attention. The particular version we use is a leaky capacitor model developed in the ACT* theory [1] and studied parameterically by Huberman and Hogg [4]. An activation network can be represented as a graph defined by matrix \mathbf{R} , where each off-diagonal element $\mathbf{R}_{i,j}$ contains the strength of association between nodes i and j , and the diagonal contains zeros. The strengths determine how much activation flows from node to node. The set of source nodes of activation being pumped into the network is represented by a vector \mathbf{C} ,

where \mathbf{C}_i represents the activation pumped in by node i . The dynamics of activation can be modeled over discrete steps $t = 1, 2, \dots, N$, with activation at step t represented by a vector $\mathbf{A}(t)$, with element $\mathbf{A}(t, i)$ representing the activation at node i at step t . The time evolution of the flow of activation is determined by

$$\mathbf{A}(t) = \mathbf{C} + \mathbf{M} \mathbf{A}(t - 1), \quad (2)$$

where \mathbf{M} is a matrix that determines the flow and decay of activation among nodes. It is specified by

$$\mathbf{M} = (1 - \gamma) \mathbf{I} + \alpha \mathbf{R}, \quad (3)$$

where $\gamma < 1$ is a parameter determining the relaxation of node activity back to zero when it receives no additional activation input, and α is a parameter denoting the amount of activation spread from a node to its neighbors. \mathbf{I} is the identity matrix.

Huberman and Hogg showed that the characteristic dynamically behavior of spreading activation depends on the relation among γ , α , and the mean number of arcs per node, μ . In the general case, there is a phase transition

Table 4. Examples of Web Groups selected using spreading activation.

Activation Source	Network	Web Group Pages (No. found)
Xerox Home Page	Paths	Xerox product descriptions (10) Financial reports (6) Business Division home pages (5) General info (2) Search form (1)
Typical person	Text similarity	Group project overviews (5) Other people hotlists (4) Company info (4) Personal interests (4) Other similar people (3) Informal groups (1) Workshop attendee list (1) Wildlife award report (1) Someone else's talk (1)

when $\alpha = \gamma$. When α/γ is small, the total activation in the net rapidly rises to an asymptotic pattern and is localized in the network. When $\alpha > \gamma$, there is another phase transition at $\mu = 1$. With $\alpha > \gamma$, when the network contains sparsely connected nodes with $\mu < 1$, the total activation rises indefinitely but the pattern remains localized. Our usage pattern graphs structures are such sparse networks. With $\alpha > \gamma$, with richly connected nodes with $\mu > 1$, the total activation rises indefinitely and all parts of the network affect all others, so that inputs of activation at any node tend to create the same pattern of activation. Our text similarity graphs are richly connected graphs. Given this characterization of the phase space of spreading activation regimes, we chose parameters such that $\alpha/\gamma \ll 1$ to identify Web structure aggregates.

To illustrate, consider the situation in which we identify the most frequently visited organization home page and wish to construct a Web aggregate that contains the pages most visited from that page. The most popular organization page can be identified as we did in Table 1 and the corresponding component of \mathbf{C} given a positive value, and the remaining elements set to zero. Setting the association matrix \mathbf{R} to be the usage flow matrix, we then iterate Equation 2 for N time steps (in our simulations we used $N = 10$). Selecting the 25 most active pages constructs the collection described in Table 5. Consider another situation in which we are interested in the Web pages having the highest text similarity to the most typical person page in a Web locality. In other words, we might be interested in understanding something about the character of the most typical person publishing in a Web

locality. In this case, the most typical person page is identified as in Table 1, the corresponding \mathbf{C} element set to positive activation input (zeros elsewhere), and \mathbf{R} is set to the text similarity matrix. Iteration of this spread of activation for $N = 10$ time steps selects the collection described in Table 5. In either case, we could have used an activation threshold rather than a fixed set size to circumscribe a Web Group.

Because of the simple properties of our activation networks, it is easy to combine the spread of activation through any weighted combination of activation pumped from different sources and through different kinds of arc--that is, simultaneously through the connectivity, usage, and text similarity connections. Consequently, the Web locality can be lit up from different directions and using different colors of relevancy. Figure 1 presents one such combined degree-of-interest pattern. Figure 1 depicts the evolution of activation over $N = 10$ time steps with the x-dimension columns representing the activation of individual Web pages. In this case, \mathbf{C} has been set to the frequency of use of Web pages and \mathbf{R} set to the text similarity associations. The resulting activation pattern is thus a combination of the most textually similar pages to the most popular pages at the Web locality.

SUMMARY

Previous work has focused on attempts to extract higher level abstractions which can be used to improve navigation and assimilation of hypertext. This research has typically used topological or textual relationships to drive analysis. In this paper, we have exploited new sources of information including usage statistics and page meta-information to develop new techniques for node typing, group extraction, and relevancy determination. These methods can provide leverage for overcoming the current usability pitfalls in user interactions with the Web.

Acknowledgments

Order of authors is alphabetical.

REFERENCES

1. Anderson, J.R. and P.L. Pirolli, Spread of activation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 10 (1984). 791-798.
2. Botafogo, R.A. and B. Schneiderman. Identifying aggregates in hypertext structures. in *Third ACM Conference on Hypertext*. (1991, San Antonio, TX).
3. Catledge, L. and J. Pitkow, Characterizing browsing strategies in the World-Wide Web. *Computer Networks and ISDN Systems*, 27 (1995). .
4. Huberman, B.A. and T. Hogg, Phase transitions in artificial intelligence systems. *Artificial Intelligence*, 33 (1987). 155-171.
5. Mukherjea, S., J.D. Foley, and S. Hudson. Visualizing complex hypermedia networks through multiple hierarchical views. in *Human Factors in Computing Systems CHI-95*. (1995, Denver, CO). ACM. pp. 331-337.
6. Pitkow, J. and C. Jehow. Results from the Third WWW Survey. in *4th Annual International WWW Conference*. (1995)
7. Robertson, G.G., S.K. Card, and J.D. Mackinlay, Information visualization using 3D interactive animation. *Communications of the ACM*, 36 (1993). 57-71.
8. Taubes, G., Indexing the internet. *Science*, 8 (1995).

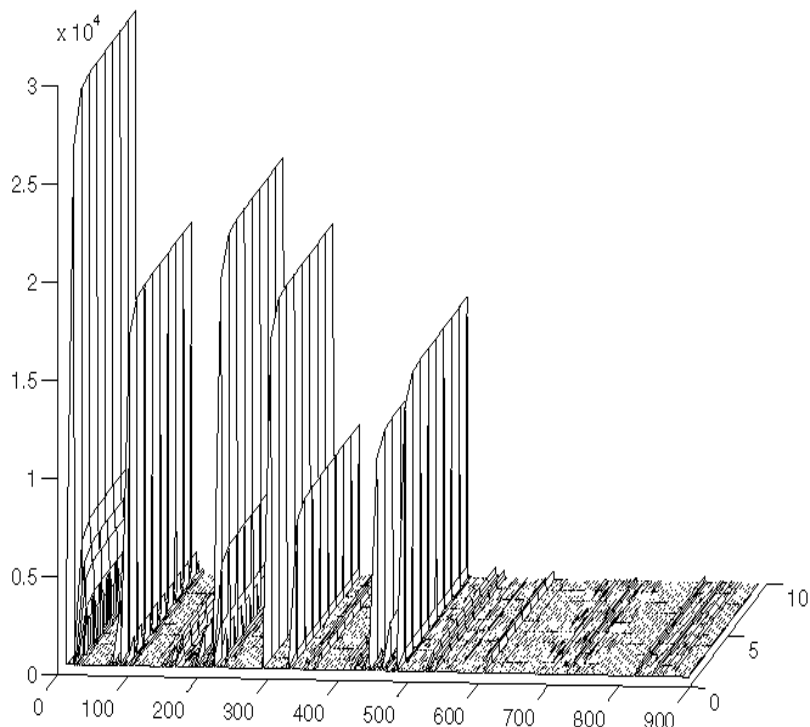


Figure 1. The evolution of activation with input sources proportion to frequency of visits and network strengths set to inter-page text similarities. The x-dimension codes individual nodes, the y-dimension, level of activation, and the z-dimension is time.

1354-1356.

9. vanRijsbergen, C.J., *Information retrieval*.
Butterworth & Co., Boston, MA, 1979.