Automatically generating hypertext in newspaper articles by computing semantic relatedness

Stephen J. Green
Microsoft Research Institute
School of Mathematics, Physics, Computing and Electronics*
Macquarie University
Sydney, NSW 2109
Australia
sjgreen@mri.mq.edu.au

Abstract

We discuss an automatic method for the construction of hypertext links within and between newspaper articles. The method comprises three steps: determining the lexical chains in a text, building links between the paragraphs of articles, and building links between articles. Lexical chains capture the semantic relations between words that occur throughout a text. Each chain is a set of related words that captures a portion of the cohesive structure of a text. By considering the distribution of chains within an article, we can build links between the paragraphs. By computing the similarity of the chains contained in two different articles, we can decide whether or not to place a link between them. We also describe the results of an evaluation performed to test the methodology.

1 Introduction

A survey, reported in Outing (1996), found that there were 1,115 commercial newspaper online services worldwide, 94% of which were on the World-Wide Web (WWW). Of these online newspapers, 73% are in North America. Outing predicted that the number of newspapers online would increase to more than 2,000 by the end of 1997.

The problem is that these services are not making full use of the hypertext capabilities of the WWW. The user may be able to navigate to a particular article in the current edition of an online paper by using hypertext links, but they must then read the entire article to find the information that interests them. These databases are "shallow" hypertexts; the documents that are being retrieved are dead ends in the hypertext, rather than offering starting points for explorations. In order to truly reflect the hypertext nature of the Web, links should to be placed within and between the documents.

As Westland (1991) has pointed out, manually creating and maintaining the sets of links needed for a large-scale hypertext is prohibitively expensive. This is especially true for newspapers, given the volume of articles produced every day. This could certainly account for the state of current WWW newspaper efforts. Aside from the time-and-money aspects of building such large hypertexts manually, humans are inconsistent in assigning hypertext links between the paragraphs of documents (Ellis et al., 1994; Green, 1997). That is, different linkers disagree with each other as to where to insert hypertext links into a document.

The cost and inconsistency of manually constructed hypertexts does not necessarily mean that large-scale hypertexts can never be built. It is well known in the IR community that humans are inconsistent in assigning index terms to documents, but this has not hindered the construction of automatic indexing systems intended to be used for very large collections of documents. Similarly, we can turn to automatically constructed hypertexts to address the issues of cost and inconsistency.

In this paper, we will describe a novel method for building hypertext links within and between newspaper articles. We have selected newspaper articles for two main reasons. First, as we stated above, there is a growing number of services devoted to providing this information in a hypertext environment. Second, many newspaper articles have a standard structure that we can exploit in building hypertext links.

Most of the proposed methods for automatic hypertext construction rely on term repetition. The underlying philosophy of these systems is that texts that are related will tend to use the *same* terms. Our system is based on *lexical chaining* and the philosophy that texts that are related will tend to use *related* terms.

2 Lexical chains

A *lexical chain* (Morris and Hirst, 1991) is a sequence of semantically related words in a text. For example, if a text contained the words *apple* and *fruit*, they would appear in a chain together, since *apple* is a kind of *fruit*. Each word in a text may appear in only one chain, but a document will contain many chains, each of which captures a portion of the cohesive structure of the document. Cohesion

*Work done at the Department of Computer Science of the University of Toronto

Stephen J. Green (1998) **Automatically generating hypertext in newspaper articles by computing semantic relatedness.** In D.M.W. Powers (ed.) *NeMLaP3/CoNLL98: New Methods in Language Processing and Computational Natural Language Learning,* ACL, pp 101-110.

is what, as Halliday and Hasan (1976) put it, helps a text "hang together as a whole". The lexical chains contained in a text will tend to delineate the parts of the text that are "about" the same thing. Morris and Hirst showed that the organization of the lexical chains in a document mirrors, in some sense, the discourse structure of that document.

The lexical chains in a text can be identified using any lexical resource that relates words by their meaning. Our current lexical chainer (based on the one described by St-Onge, 1995) uses the WordNet database (Beckwith et al., 1991). The WordNet database is composed of synonym sets or *synsets*. Each synset contains one or more words that have the same meaning. A word may appear in many synsets, depending on the number of senses that it has. Synsets can be connected to each other by several different types of links that indicate different relations. For example, two synsets can be connected by a hypernym link, which indicates that the words in the source synset are instances of the words in the target synset.

For the purposes of chaining, each type of link between WordNet synsets is assigned a direction of up, down, or horizontal. Upward links correspond to generalization: for example, an upward link from *apple* to *fruit* indicates that *fruit* is more general than *apple*. Downward links correspond to specialization: for example, a link from *fruit* to *apple* would have a downward direction. Horizontal links are very specific specializations. For example, the antonymy relation in WordNet is given a direction of horizontal, since it specializes the sense of a word very accurately, that is, if a word and its antonym appear in a text, the two words are very likely being used in the senses that are antonyms.

Given these types of links, three kinds of relations are built between words:

Extra strong An extra strong relation is said to exist between repetitions of the same word: i.e., term repetition.

Strong A strong relation is said to exist between words that are in the same WordNet synset (i.e., words that are synonymous). Strong relations are also said to exist between words that have synsets connected by a single horizontal link or words that have synsets connected by a single IS-A or INCLUDES relation.

Regular A regular relation is said to exist between two words when there is at least one *allowable* path between a synset containing the first word and a synset containing the second word in the WordNet database. A path is allowable if it is short (less than n links, where n is typically 3 or 4) and adheres to three rules:

1. No other direction may precede an upward link.
2. No more than one change of direction is allowed.
3. A horizontal link may be used to move from an upward to a downward direction.

When a word is processed during chaining, it is initially associated with all of the synsets of which it is a member. When the word is added to a chain, the chainer attempts to find connections between the synsets associated with the new word and the synsets associated with words that are already in the chain. Synsets that can be connected are retained and all others are discarded. The result of this processing is that, as the chains are built, the words in the chains are progressively sense-disambiguated. When an article has been chained, a description of the chains contained in the document is written to a file. Table 1 shows some of the chains that were recovered from an article about the trend towards "virtual parenting" (Shellenbarger, 1995). In this table, the numbers in parentheses show the number of occurrences of a particular word.

The process of lexical chaining is not perfect, but if we wish to process articles quickly, then we must accept some errors or at least bad decisions. In our sample article, for example, chain 1 is a conglomeration of words that would have better been separated into different chains. This is a side effect of the current implementation of the lexical chainer, but even with these difficulties, we are able to perform useful tasks. We expect to address some of these problems in subsequent versions of the chainer, hopefully with no loss in efficiency.

3 Building links within an article

3.1 Analyzing the lexical chains

Newspaper articles are written so that one may stop reading at the end of any paragraph and feel as though one has read a complete unit. For this reason, it is natural to choose to use paragraphs as the nodes in our hypertext. Table 1 showed the lexical chains recovered from a news article about the trend towards "virtual parenting". Figure 1 shows the second and eighth paragraphs of this article with the words that participate in lexical chains tagged with their chain numbers. We will use this particular article to illustrate the process of building intra-article links.

The first step in the process is to determine how important each chain is to each paragraph in an article. We judge the importance of a chain by calculating the fraction of the content words of the paragraph that are in that chain. We refer to this fraction as the *density* of that chain in that paragraph. The density of chain c in paragraph p, $d_{c,p}$, is defined as:

$$d_{c,p} = \frac{w_{c,p}}{w_p}$$

Table 1: Some lexical chains from the virtual parenting article.

C	Word	Syn	C	Word	Syn	C	Word	Syn
1	working (5)	40755		expert (1)	59108	12	giving (1)	19911
	ground (1)	58279		mark (1)	60270		pushing (1)	20001
	field (1)	57992		worker (1)	59145		push (1)	20001
	antarctica (1)	58519		speaker (1)	63258		high-tech (2)	19957
	michigan (1)	57513		advertiser (1)	59643	19	planning (1)	23089
	feed (1)	53429		entrepreneur (1)	60889		arranging (1)	23127
	chain (1)	57822		engineer (1)	59101	21	good_night (1)	48074
	hazard (1)	77281		sitter (1)	59827		wish (1)	48061
	risk (1)	77281		consultant (2)	59644	22	phone (2)	40017
	young (2)	24623		management_consultant (1)	61903		cellular_phone (1)	33808
	need (1)	58548		man (1)	61902		fax (2)	35302
	parent (7)	62334		flight_attendant (1)	63356		gear (1)	32030
	kid (3)	60256	4	folk (1)	54362		joint (2)	36574
	child (1)	60256		family (4)	54362		junction (1)	36604
	baby (1)	59820	10	management (2)	55578		network (1)	37247
	wife (1)	63852		professor (1)	62638		system (2)	32196
	adult (1)	59073		conference (1)	55372		audiotape (1)	39983
	traveller (3)	59140		meeting (1)	55371		gadget (1)	32428
	substitute (1)	63327		school (1)	55261	23	feel (1)	22808
	backup (1)	63327		university (1)	55299		kissing (1)	22806
	computer (1)	60118		company (1)	54918			

> Although no one is **pushing**[12] virtual-reality **headgear**[16] as a **substitute**[1] for **parents**[1], many technical ad **campaigns**[13] are promoting cellular **phones**[22], **faxes**[22], **computers**[1] and pagers to **working**[1] **parents**[1] as a way of bridging **separations**[17] from their **kids**[1]. A recent **promotion**[13] by A T & T and **Residence**[2] **Inns**[7] in the **United States**[6], for **example**[3], suggests that **business**[3] **travellers**[1] with **young**[1] children use **video**[3] and **audio tapes**[22], **voice**[3] **mail**[3], videophones and E-mail to **stay**[3] connected, including **kissing**[23] the **kids**[1] **good night**[21] by **phone**[22].

> More **advice**[3] from **advertisers**[1]: **Business**[3] **travellers**[1] can dine with their **kids**[1] by **speaker**[1]-phone or "tuck them in" by cordless **phone**[22]. Separately, a **management**[10] **newsletter**[24] recommends faxing your **child**[1] when you have to **break**[17] a **promise**[3] to be **home**[2] or **giving**[12] a **young**[1] **child**[1] a beeper to make him **feel**[23] more secure when **left**[5] alone.

Figure 1: Two portions of a text tagged with chain numbers.

where $w_{c,p}$ is the number of words from chain c that appear in paragraph p and w_p is the number of content words (i.e., words that are not stop words) in p. For example, if we consider paragraph two of our sample article, we see that there are 9 words from chain 1. We also note that there are 48 content words in the paragraph. So, in this case the density of chain 1 in paragraph 1, $d_{1,2}$, is $\frac{9}{48} = 0.19$.

The result of these calculations is that each paragraph in the article has associated with it a vector of chain densities, with an element for each of the chains in the article. Table 2 shows these *chain density vectors* for the chains shown in table 1. Note that an empty element indicates a density of 0.

3.2 Determining paragraph links

As we said earlier, the parts of a document that are about the same thing, and therefore related, will tend to contain the same lexical chains. Given the chain density vectors that we described above, we need to develop a method to determine the similarity of the sets of chains contained in each paragraph. The second stage of paragraph linking, therefore, is to compute the similarity between the paragraphs of the article by computing the similarity between the chain density vectors representing them. We can compute these similarities using any one of 16 similarity coefficients that we have taken from Ellis et al. (1994).

This similarity is computed for each pair of chain density vectors, giving us a symmetric $p \times p$ matrix of similarities, where p is the number of paragraphs in the article. From this matrix we can calculate the mean and the standard deviation of the paragraph similarities.

The next step is to decide which paragraphs should be linked, on the basis of the similarities computed in the previous step. We make this decision by looking at how the similarity of two paragraphs compares to the mean paragraph similarity across the entire article. Each similarity between two paragraphs i and j, $s_{i,j}$, is converted

Table 2: Some chain density vectors for the virtual parenting article.

	Paragraph										
Chain	1	2	3	4	5	6	7	8	9	10	11
1	0.14	0.19	0.07	0.16	0.28	0.18	0.10	0.25	0.24	0.13	0.33
4	0.07		0.11	0.05			0.03			0.03	
10			0.07	0.05		0.11		0.04		0.03	
12		0.02	0.04	0.05				0.04	0.03		
19			0.04		0.06						
21		0.02		0.05							
22		0.08	0.04	0.05	0.11		0.07	0.07	0.08	0.03	
23		0.02						0.04			
Chain Words	8	30	15	15	10	15	16	19	20	15	6
Content	14	48	27	19	18	28	29	28	38	30	9
Density	0.57	0.62	0.56	0.79	0.56	0.54	0.55	0.68	0.53	0.50	0.67

Table 3: Adjacency matrix for the virtual parenting article.

Par	1	2	3	4	5	6	7	8	9	10	11
1	0	0	0	0	0	0	0	0	0	0	0
2		0	0	0	1	0	0	1	1	1	0
3			0	0	0	0	0	0	0	0	0
4				0	0	0	0	0	0	0	0
5					0	0	0	1	1	0	0
6						0	0	0	0	1	0
7							0	0	0	1	0
8								0	1	0	0
9									0	0	1
10										0	0
11											0

to a z-score, $z_{i,j}$. If two paragraphs are more similar than a threshold given in terms of a number of standard deviations, then a link is placed between them. The result is a symmetric adjacency matrix where a 1 indicates that a link should be placed between two paragraphs. Figure 3 shows the adjacency matrix that is produced when a z-score threshold of 1.0 is used to compute the links for our virtual parenting example.

Once we have decided which paragraphs should be linked, we need to be able to produce a representation of the hypertext that can be used for browsing. In the current system, there are two ways to output the HTML representation of an article. The first simply displays all of the links that were computed during the last stage of the process described above. The second is more complicated, showing only some of the links. The idea is that links between physically adjacent paragraphs should be omitted so that they do not clutter the hypertext.

4 Building links between articles

While it is useful to be able to build links *within* articles, for a large scale hypertext, links also need to be placed *between* articles. You will recall from section 2 that the output of the lexical chainer is a list of chains, each chain consisting of one or more words. Each word in a chain has associated with it one or more synsets. These synsets indicate the sense of the word as it is being used in this chain. An example of the kind of output produced by the chainer is shown in table 4, which shows a portion of the chains extracted from an article (Gadd, 1995b) about cuts in staff at children's aid societies due to a reduction in provincial grants. Table 5 shows a portion of another set of chains, this time from an article (Gadd, 1995a) describing the changes in child-protection agencies, due in part to budget cuts.

It seems quite clear that these two articles are related, and that we would like to place a link from one to the other. It is also clear that the words in these two articles display both of the linguistic factors that affect IR performance, namely synonymy and polysemy. For example, the first set of chains contains the word *abuse*, while the second set contains the synonym *maltreatment*. Similarly, the first set of chains includes the word *kid*, while the second contains *child*. The word *abuse* in the first article has been disambiguated by the lexical chainer into the "cruel or inhuman treatment" sense, as has the word *maltreatment* from the second article. We once again note that the lexical chaining process is not perfect: for example, both texts contain the word *abuse*, but it has been disambiguated into different senses — in the first article, it is meant in the sense of "ill-treatment", while in the second it is meant in the sense of "verbal abuse".

Although the articles share a large number of words, by missing the synonyms or by making incorrect (or no) judgments about different senses, a traditional IR system might miss the relation between these documents or rank them as less related than they really are. Aside from the problems of synonymy and polysemy, we can see that there are also more-distant relations between the words of these two articles. For example, the second set of chains

Table 4: Some lexical chains from an article about cuts in children's aid societies.

C	Word	Syn	C	Word	Syn	C	Word	Syn
3	society (7)	54351		annual (1)	64656		care (1)	22204
	group (1)	19698	5	ontario (1)	56918		social_work (1)	24180
	mother (1)	62088		canadian (1)	58424		slowdown (1)	23640
	parent (4)	62334			59296		abuse (3)	21214
	kid (1)	60256		burlington (1)	57612		child_abuse (1)	21215
	recruit (1)	62769		union (3)	57424		neglect (1)	21235
	employee (2)	60862	10	saying (1)	50294	28	living (1)	75629
	worker (2)	59145		interview (2)	50268		standing (1)	75573
	computer (1)	60118	27	try (1)	22561		complaint (1)	76270
	teen-ager (2)	59638		seeking (1)	22571		agency (1)	75786
	provincial (3)	62386		acting (1)	21759		stress (1)	76799
	face (1)	59111		services (1)	21922			76906
	spokesman (1)	63287		work (3)	21919	32	executive_director (2)	60922
	insolvent (1)	59869		risk (2)	22613		manager (1)	59634

Table 5: Some lexical chains from a related article.

C	Word	Syn	C	Word	Syn	C	Word	Syn
2	wit (1)	48647		guardian (1)	59099			24236
	play (1)	48668		official (1)	62223		making (1)	23076
	abuse (4)	48430		worker (1)	59145		calling (1)	21911
	cut (4)	48431		neighbour (1)	62152		services (2)	21922
	criticism (1)	48406		youngster (1)	60255		prevention (1)	23683
	recommendation (1)	48310		kid (2)	60255		supply (1)	23596
	case (1)	48682		natural (1)	62139		providing (3)	23596
	problem (1)	48680		lawyer (2)	61725		maltreatment (2)	21214
	question (3)	48679		professional (1)	62636		child_abuse (2)	21215
3	child (10)	60256		prostitute (1)	62660		investigation (1)	22142
	parent (9)	62334		provincial (2)	62386		research (1)	22143
	mother (3)	62088		welfare_worker (1)	63220		investigating (1)	22142
	daughter (1)	60587		lorelei (1)	61833		work (1)	21885
	foster_home (1)	54374		god (1)	58615		aid (9)	22204
	society (5)	54351	4	protection (2)	22672		social_work (1)	24180
	at_home (1)	55170		care (5)	22721		risk (1)	22613
	social (1)	55184		preservation (2)	22676		dispute (1)	24051
	function (1)	55154		judgment (1)	22881		intervention (1)	24317
	expert (3)	59108		act (1)	19697		fail (1)	19811
	human (1)	19677		behaviour (1)	24235			

contains the word *maltreatment* while the first set contains the related word *child abuse* (a kind of maltreatment) as well as the repetition of *child abuse*.

We can build these *inter-article* links by determining the similarity of the two *sets* of chains contained in two articles. In essence, we wish to perform a kind of cross-document chaining.

4.1 Synset weight vectors

We can represent each document in a database by two vectors. Each vector will have an element for each synset in WordNet. An element in the first vector will contain a weight based on the number of occurrences of that particular synset in the words of the chains contained in the document. An element in the second vector will contain a weight based on the number of occurrences of that particular synset when it is one link away from a synset associated with a word in the chains. We will call these vectors the *member* and *linked synset vectors*, or simply the member and linked vectors, respectively.

The weight of a particular synset in a particular document is not based solely on the frequency of that synset in the document, but also on how frequently that term appears throughout the database. The synsets that are the most heavily weighted in a document are the ones that appear frequently in that document but infrequently in the entire database. The weights are calculated using the standard tf·idf weighting function:

$$w_{ik} = \frac{sf_{ik} \cdot \log(N/n_k)}{\sqrt{\sum_{j=1}^{s}(sf_{ij})^2 \cdot (\log(N/n_j))^2}}$$

where sf_{ik} is the frequency of synset k in document i, N is the size of the document collection, n_k is the number of documents in the collection that contain synset k, and s is the number of synsets in all documents. Note that this equation incorporates the normalization of the synset weight vectors.

The weights are calculated independently for the member and linked vectors. We do this because the linked vectors introduce a large number of synsets that do not necessarily appear in the original chains of an article, and should therefore not influence the frequency counts of the member synsets. Thus, we make a distinction between strong links that occur due to synonymy, and strong links that occur due to IS-A or INCLUDES relations. The similarity between two documents, D_1 and D_2, is then determined by calculating three cosine similarities:

1. The similarity of the member vectors of D_1 and D_2;
2. The similarity of the member vector of D_1 and linked vector of D_2; and
3. The similarity of the linked vector of D_1 and the member vector of D_2.

Clearly, the first similarity measure (the *member-member* similarity) is the most important, as it will capture extra-strong relations as well as strong relations between synonymous words. The last two measures (the *member-linked* similarities) are less important as they capture strong relations that occur between synsets that are one link away from each other. If we enforce a threshold on these measures of relatedness, then we ensure that there are several connections between two articles, since each element of the vectors will contribute only a small part of the overall similarity.

4.2 Building inter-article links

Once we have built a set of synset weight vectors for a collection of documents, the process of building links between articles is relatively simple. Given an article that we wish to build links from, we can compute the similarity between the article's synset weight vectors and the vectors of all other documents. Documents whose member vectors exceed a given threshold of similarity will have a link placed between them. Our preliminary work shows that a threshold of 0.15 will include most related documents while excluding many unrelated documents.

This is almost exactly the methodology used in vector-space IR systems such as SMART, with the difference being that for each pair of documents we are calculating three separate similarity measures. The best way to cope with these multiple measurements seems to be to rank related documents by the sum of the three similarities. The sum of the three similarities can lie, theoretically, anywhere between 0 and 3. In practice, the sum is usually less than 1. For example, the average sum of the three similarities when running the vectors of a single article against 5,592 other articles is 0.039.

5 Evaluation

In the evaluation that we conducted, the basic question that we asked was: Is our hypertext linking methodology superior to other methodologies that have been proposed (e.g., that of Allan, 1995)? The obvious way to answer the question was to test whether the links generated by our methodology lead to better performance when they were used in the context of an appropriate IR task.

We selected a question-answering task for our study. We made this choice because it appears that this kind of task is well suited to the browsing methodology that hypertext links are meant to support. This kind of task is also useful because it can be performed easily using *only* hypertext browsing. This is necessary because in the interface used for our experiment, no query engine was provided for the subjects.

We used the "Narrative" section of three TREC topics (Harman, 1994) to build three questions for our subjects to answer. There were approximately 1996 documents that were relevant to the topics from which these questions were created. We read these documents and prepared lists of answers for the questions. Our test database consisted of these articles combined randomly with approximately 29,000 other articles selected randomly from the TREC corpus. The combination of these articles provided us with a database that was large enough for a reasonable evaluation and yet small enough to be easily manageable.

5.1 The test system

We considered two possible methods for generating inter-article hypertext links. The first is our own method, described above. The second method uses a vector space IR system called Managing Gigabytes (MG) (Witten et al., 1994) to generate links by calculating a document similarity that is based strictly on term repetition. We used the MG system to generate links in a way very similar to that presented in Allan (1995). For simplicity's sake, we will call the links generated by our technique *HT links* and the links generated by the MG system *MG links*.

Figure 2 shows the interface of the test system used. The main part of the screen showed the text of a single article. The subjects could navigate through the article by using the intra-article links, a scroll bar, or the page up and down keys. The *Previous Article* and *Next Article* buttons could be used for navigating through the set of articles that had been visited and the *Back* button returned the user to the point from which an intra-article link was taken. Each search began on a "starter" page that contained the text of the appropriate TREC topic as the "article" and the list of articles related to the topic shown (this was computed by using the text of the topic as the initial "query" to the database). Subjects were expected to traverse the links, writing down whatever answers they could find.

At each stage during a subject's browsing, a set of inter-article links was generated by combining the set of HT links and the set of MG links. By using this strategy, the subjects "vote" for the system that they prefer by choosing the links generated by that system. Of course, the subjects are not aware of which system generated the links that they are following — they can only decide to

Figure 2: The interface of the evaluation system.

follow a link by considering the article headlines displayed as anchors. We can, however, determine which system they "voted" for by considering their success in answering the questions they were asked. If we can show that their success was greater when they followed more HT links, then we can say that they have "voted" for the superiority of HT links. A similar methodology has been used previously by Nordhausen et al. (1991) in their comparison of human and machine-generated hypertext links.

The two sets of inter-article links can be combined by simply taking the *unique* links from each set, that is, the links that we take are those that appear in only one of the sets of links. Of course, we would expect the two methods to have many links in common, but it is difficult to tell how these links should be counted in the "voting" procedure. By leaving them out, we test the differences between the methods rather than their similarities. Of course, by excluding the links that the methods agree on we are reducing the ability of the subjects to find answers to the questions that we have posed for them. In fact, we found that nearly 40% of the links found were found by both methods. It does seem, however, that the users could find enough answers to give some interesting results.

5.2 Experimental results

The number of both inter- and intra-article links followed was, on average, quite small and variable (full data are given in Green, 1997). The number of correct answers found was also low and variable, which we believe is due partly to the methodology and partly to the time restrictions placed on the searches (15 minutes). On average, the subjects showed a slight bias for HT links, choosing 47.9% MG links and 52.1% HT links. This is interesting, especially in light of the fact that, for all the articles the subjects visited, 50.4% of the links available were MG links, while 49.6% were HT links. A paired t-test, however indicates that this difference is not significant.

For the remainder of the discussion, we will use the variable L_{HT} to refer to the number of HT links that a subject followed, L_{MG} to refer to the number of MG links followed, and L_I to refer to the number of intra-article links followed. The variable Ans will refer to the number of correct answers that a subject found. We can combine L_{HT} and L_{MG} into a ratio, $L_R = \frac{L_{HT}}{L_{MG}}$. If $L_R > 1$, then a subject followed more HT links than MG links. An interesting question to ask is: did subjects with significantly higher values for L_R find more answers? With 23 subjects each answering 3 questions, we have 69 values for L_R. If we sort these values in decreasing order and divide the resulting list at the median, we have two groups with a significant difference in L_R. An unpaired t-test then tells us that the differences in Ans for these two groups are significant at the 0.1 level.

So it seems that there may be some relationship between the number and kinds of links that a subject followed and his or her success in finding answers to the questions pose. We can explore this relationship using two different regression analyses, one incorporating only inter-article links and another incorporating both inter- and intra-article links. These analyses will express the relationship between the number of links followed and the number of correct answers found.

5.2.1 Inter-article links

A model incorporating only the inter-article links that our subjects followed gives us the following equation:

$$Ans = 0.46 \cdot L_{HT} + 0.17 \cdot L_{MG} \qquad (R^2 = 0.09)$$

which shows a greater benefit (in terms of the number of answers found) for the selection of an HT link over an MG link. An ANOVA analysis of this model hows that our independent variables are related to our independent variable and that with $p \leq 0.05$, we can safely assume that the number of links followed is related to the number of answers found.

The 95% confidence intervals for the model coefficients are shown in table 6. Here, the column labeled t is the t-score associated with the hypothesis H_0: the coefficient in question is 0. The alternative hypothesis is that the coefficient is greater than 0. The column labeled p is the probability that H_0 is true. The columns labeled *Low* and *High* give the endpoints of the 95% confidence interval for the values of each of the coefficients.

Notice that there is a small overlap between the confidence intervals for the two coefficients. Thus we cannot reject our null hypothesis that there is no difference in benefit from following an HT link versus an MG link. By inspection, we find that the confidence intervals begin overlapping at approximately the 92.5% level.

Table 6: 95% confidence intervals for inter-article links.

Parameter	Value	t	p	Low	High
L_{HT}	0.46	5.96	0.00	0.31	0.62
L_{MG}	0.17	2.01	0.02	0.00	0.34

We can use our ratio measure, L_R to visualize the data set in two dimensions, as in figure 3. Table 7 shows the 95% confidence intervals for the parameters of this model. From this table, we see that we can reject the hypothesis that the coefficient of L_R is 0 with $p < 0.05$. The 95% confidence interval for this coefficient is not entirely positive, which indicates that a some points there may be a greater benefit from following MG links.

Table 7: 95% confidence intervals for a two-dimensional model.

Parameter	Value	t	p	Low	High
Constant	3.65	6.52	0.00	2.53	4.77
L_R	0.56	1.90	0.03	−0.03	1.16

5.2.2 Inter- and intra-article links

When we include the intra-article links in our analysis, we obtain the following model:

$$Ans = 0.44 \cdot L_{HT} + 0.15 \cdot L_{MG} + 0.06 \cdot L_I \qquad (R^2 = 0.10)$$

As with the model discussed above, there is still a greater benefit in selecting an HT link over an MG link. The coefficient of L_I, although quite small, is positive, indicating some benefit from following intra-article links. The ANOVA analysis for this model indicates that our independent variables are indeed related to our dependent variables. The 95% confidence intervals of the model coefficients in table 8 show that, as with the models discussed above, we cannot reject our null hypothesis with respect to the inter-article links. Also, we note that the probability that the coefficient of L_I is 0 is quite high ($p > 0.18$).

Table 8: 95% confidence intervals for inter- and intra-article links.

Parameter	Value	t	p	Low	High
L_{HT}	0.44	5.55	0.00	0.28	0.60
L_{MG}	0.15	1.70	0.05	−0.03	0.32
L_I	0.06	0.92	0.18	−0.07	0.18

Thus we are lead to conclude that intra-article links had no across-the-board effect on *Ans* for this particular question-answering task.

5.2.3 Data by experience

We can also ask how a subject's success is affected by their degree of previous experience in using hypertext. We divide the subjects into two groups. The first group, which we will call the *Low Web* group use the World Wide Web less than 3 times a week, while the second group (the *High Web* group) use the Web 3 or more times a week. An unpaired t-test shows that the High Web group (12 subjects), on average, chose significantly more ($p < 0.01$) inter-article links than the Low Web group (11 subjects). This difference indicates that these subjects are probably more comfortable in a hypertext environment, and adapted more quickly to the interface used for the task.

When we look at the numbers of each kind of hypertext links followed by each group, we see that the High Web group chose significantly more HT links than the Low Web group ($p < 0.01$). There was no significant difference in the number of MG links chosen by the two groups. Within each group, we find that the High Web group chose significantly ($p < 0.05$) more HT links than MG links, while there was no such significant difference in the Low Web group. There is also a significant difference ($p < 0.01$) in the number of answers found by the two groups, with the High Web group finding more correct answers.

If we consider the inverse of our ratio measure, $\frac{1}{L_R}$, then we see a significant ($p < 0.05$) difference in the ratios between the High and Low Web groups. Thus,

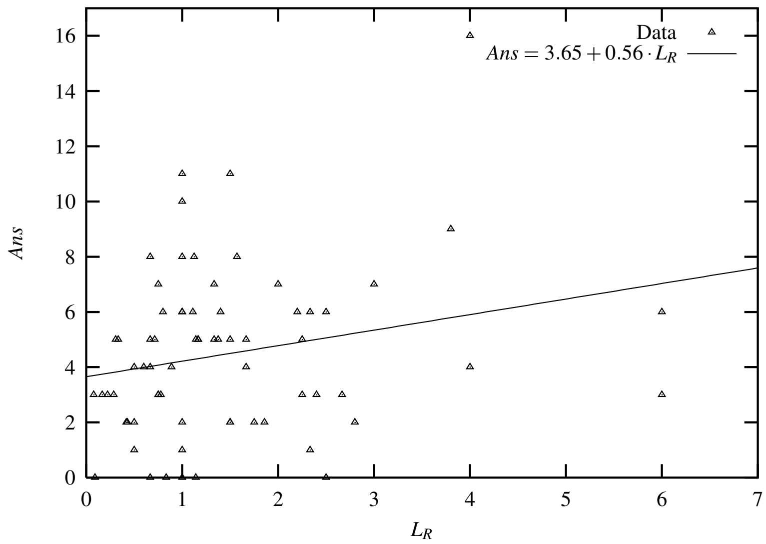

Figure 3: Data and regression line for a two-dimensional model.

we can see a set of subjects (the High Web group) who found significantly more answers *and* followed significantly more HT links, indicating the advantage of HT links over MG links.

5.2.4 Viewed answers

In the analyses that we've performed to this point, we have been using the number of correct answers that the subjects provided as our dependent variable. Part of the reason we are using this dependent variable is that the subjects were limited in the amount of time that they could spend on each search, and so they could only find a certain number of answers, no matter how many answers there were to find. We can mitigate this effect by introducing a new dependent variable, Ans_V, or the number of *viewed answers*.

The number of viewed answers for a particular question is simply the number of answers that were contained in articles that a subject visited while attempting to answer a question. These answers need not have been written down. We are merely saying that, given more time, the subjects might have been able to read the article more fully and find these answers. This idea is analogous to the use of *judged* and *viewed recall* by Golovchinsky (1997) in his studies.

When we consider Ans_V as our dependent variable, the model for the High Web group is still not significant, and there is still a high probability that the coefficient of L_I is 0. For our Low Web group, who followed significantly more intra-article links than the High Web group, the model that results is significant and has the following equation:

$$Ans_V = 0.58 \cdot L_{HT} + 0.21 \cdot L_{MG} + 0.21 \cdot L_I \qquad (R^2 = 0.41)$$

Table 9: 95% confidence intervals for coefficients in a model using viewed answers.

Parameter	Value	t	p	Low	High
L_{HT}	0.58	4.37	0.00	0.31	0.85
L_{MG}	0.21	1.62	0.06	−0.05	0.47
L_I	0.21	2.19	0.02	0.01	0.40

Table 9 shows the 95% confidence intervals for this model. We see that the coefficient of L_I is always positive, indicating some effect on Ans_V from intra-article links. We also see that the probability that this coefficient is 0 is less than 0.02. We note, however, that for this model we cannot claim that the coefficient of L_{HT} is always greater than the coefficient of L_{MG}. This is not too surprising in light of the fact that the High Web group chose significantly more HT links than did the Low Web group.

6 Conclusions and future work

Our evaluation shows that we cannot reject our null hypothesis that there is no difference in the two methods for generating inter-article links. Having said this, we can demonstrate a partition of the subjects such that the only significant differences between them are the number of HT links followed and the number of answers found. Furthermore, we determined that the probability of obtaining results such as these by chance is less than 0.1. Our inability to achieve a significant result may be due to several implementation factors, described in Green (1997). Thus, we conclude that we need to replicate the experiment in order to gain further information about the relationship between the two kinds of inter-article links.

Unfortunately, we also cannot say that our intra-article links are useful in all cases, although they may provide some benefit to novice users of an information system. We believe that we need to replicate this study in order to draw firmer conclusions about the method's usefulness.

One of the advantages of Allan's work (1995) in automatic hypertext generation is that the links between portions of two texts can be given a type that reflects what sort of link is about to be followed. We currently have no method for producing such typed links, but it may be the case that the relations between words from WordNet can be used to determine the type of some links.

It is still not clear how much of our methodology depends on the structure of the newspaper articles that we are processing. Does this standard structure enhance our hypertext linking capabilities, or would the method perform equally well, given any well-written text to work with? We intend to see how well the method performs on other types of texts, possibly changing our methodology to cope with the loss of some structure.

While other automatic hypertext generation methodologies have been proposed, many of them rely on term repetition to build links within and between documents. If there is no term repetition, there are no links. This is especially a problem when attempting to build intra-document links in shorter documents when an author may have been striving to avoid using the same word again and again and so chose a related word. We avoid this problem by using lexical chains, which collect words on the basis of their semantic similarity. Our results to date have shown promise for the methodology, and work is continuing.

Acknowledgments

The author wishes to thank Graeme Hirst and Lisa Chislett for their comments on earlier versions of this paper. I would also like to thank Mark Chignell and Marilyn Mantei for their invaluable help in designing the evaluation. Thanks also to the *Globe and Mail* for providing the test data. Funding for this work was provided by the Natural Sciences and Engineering Research Council of Canada and the Information Technology Research Centre of Ontario.

References

(Allan, 1995) James Allan. *Automatic hypertext construction*. PhD thesis, Cornell University, 1995.

(Beckwith et al., 1991) Richard Beckwith, Christiane Fellbaum, Derek Gross, and George A. Miller. WordNet: A lexical database organized on psycholinguistic principles. In Uri Zernik, editor, *Lexical acquisition: Exploiting on-line resources to build a lexicon*, pages 211–231. Lawrence Erlbaum Associates, 1991.

(Ellis et al., 1994) David Ellis, Jonathan Furner-Hines, and Peter Willett. The creation of hypertext linkages in full-text documents: Parts I and II. Technical Report RDD/G/142, British Library Research and Development Department, April 1994.

(Gadd, 1995a) Jane Gadd. Child aid "on double-edged sword". *The Globe and Mail*, page A14, December 5 1995.

(Gadd, 1995b) Jane Gadd. Children's aid societies plan staff, services cuts. *The Globe and Mail*, page A10, September 8 1995.

(Golovchinsky, 1997) Gene Golovchinsky. *From information retrieval to hypertext and back again: The role of interaction in the information exploration interface*. PhD thesis, University of Toronto, 1997.

(Green, 1997) Stephen J. Green. *Automatically generating hypertext by computing semantic similarity*. PhD thesis, University of Toronto, 1997.

(Halliday and Hasan, 1976) M.A.K. Halliday and Ruqaiya Hasan. *Cohesion in English*. Longman, 1976.

(Harman, 1994) Donna Harman. Overview of the third Text Retrieval Conference (TREC-3). In *Proceedings of the third Text Retrieval Conference*, November 1994.

(Morris and Hirst, 1991) Jane Morris and Graeme Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, **17**(1):21–48, 1991.

(Nordhausen et al., 1991) Bernd Nordhausen, Mark H. Chignell, and John Waterworth. The missing link? Comparison of manual and automated linking in hypertext engineering. In *Proceedings of the Human Factors Society 35th annual meeting*, 1991.

(Outing, 1996) Steve Outing. Newspapers online: The latest statistics. *Editor and Publisher Interactive [Online]*, May 13 1996. Available at: `http://www.mediainfo.com/ephome/news/newshtm/stop/stop513.htm`.

(Shellenbarger, 1995) Sue Shellenbarger. High-tech parenting virtually a finger tip away. *The Globe and Mail*, page A10, December 12 1995.

(St-Onge, 1995) David St-Onge. *Detecting and correcting malapropisms with lexical chains*. Master's thesis, University of Toronto. Published as technical report CSRI-319, 1995.

(Westland, 1991) J. Christopher Westland. Economic constraints in hypertext. *Journal of the American Society for Information Science*, **42**(3):178–184, 1991.

(Witten et al., 1994) Ian H. Witten, Alistair Moffat, and Timothy C. Bell. *Managing Gigabytes: Compressing and indexing documents and images*. Van Nostrand Reinhold, 1994.