Automated link generation: can we do better than term repetition?

Stephen J. Green

Microsoft Research Institute, School of MPCE, Macquarie University
Sydney NSW 2109, Australia
sjgreen@mri.mq.edu.au

Abstract

Most current automatic hypertext generation systems rely on term repetition to calculate the relatedness of two documents. There are well-recognized problems with such approaches, most notably, they are vulnerable to the linguistic effects of synonymy (many words for the same concept) and polysemy (many concepts for the same word). I propose a novel method for automatic hypertext generation that is based on a technique called *lexical chaining,* a method for discovering sets of related words in a text. I will also present the results of an empirical study designed to test this method in the context of a question answering task from a database of newspaper articles.

Keywords

Automatic hypertext generation; Semantic relatedness

1. Introduction

There is no question that building and maintaining a large Web site requires large amounts of time and money (Westland, 1991). Aside from these concerns, there is evidence (Green, 1997; Ellis et al., 1994) that when humans construct hypertext links they do so inconsistently, that is, different people will tend to place different links into the same document. This inconsistency may mean that the links would be less useful for a user searching for specific information. The cost and inconsistency of manually constructed hypertext links do not necessarily mean that large-scale hypertexts (e.g., a large online newspaper) can never be built, it simply means we need to turn to automatically generated hypertext links. For the most part, automatic hypertext generation in large document collections has been treated as a special case of the more general information retrieval (IR) problem. The basic premise underlying most current IR systems is that documents that are related in some way will use the *same* words. If two documents share enough terms, then we can assume that they are related and should therefore have a link placed between them.

Two linguistic factors can affect this operation: *synonymy* (many words referring to the same concept, for example, *dog* and *hound*) and *polysemy* (many concepts having the same word, for example, *bank*). The impact of synonymy is that documents that use words that are synonyms of one another will not be considered related or at best will be considered to be less related than they actually are. Polysemy will have the opposite effect, causing documents that use the same word in different senses to be considered related when they should not be.

Others have tried to account for these factors in IR systems and met with limited success (cf. Voorhees'

(1994) work on query expansion.) In this paper, I will describe a novel method for building hypertext links within and between documents. The method is intended to be a strong first step towards accounting for both synonymy and polysemy. In addition, I will propose a more general notion of relatedness than is used in traditional IR systems: Two documents will be considered related if they use *semantically related* words. The method is based on *lexical chaining*, a technique for extracting the sets of related words that occur in texts. Finally, I will describe the results of an experiment that tests the proposed hypertext generation methodology against a methodology based on a traditional IR system.

2. Lexical chaining

A *lexical chain* (Morris and Hirst, 1991) is a sequence of semantically related words in a text. For example, if a text contains the words *apple* and *fruit*, then they will appear in a chain together, since *apple* is a kind of *fruit*. The lexical chains in a document 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 the document.

The lexical chains in a text can be recovered using any lexical resource that relates words by their meanings. While the original work was done using *Roget's Thesaurus* (Chapman, 1992), the current version of the lexical chainer, similar to the one described in St-Onge (1995), uses the WordNet database (Beckwith et al., 1991). WordNet divides words up into synonym sets or *synsets*, groups of words that are synonyms of one another. These synsets are then connected by a number of different relations such as IS-A or INCLUDES. A particular word may appear in several synsets, depending on how many senses that it has.

Given the WordNet database, we can build three kinds of relations between words:

Extra Strong:
: An extra strong relation exists between repetitions of the same word.

Strong:
: A strong relation exists between words that are in the same WordNet synset (i.e., words that are synonyms of one another), or words in synsets that are connected by a single ANTONYMY, IS-A, or INCLUDES relation.

Regular:
: A regular relation exists between words when there is an *allowable* path in the WordNet graph between synsets that contain the two words. An allowable path is one that has a certain shape, and is shorter than a given (small) length.

Initially, each word is represented as a list of all the synsets that contain it. As chaining proceeds and relations are built between words, synsets that do not participate in the relations are discarded and as a result, the words in the lexical chains are progressively sense-disambiguated. Compared to traditional document processing tasks in IR (e.g., keyword extraction), the chaining process is slow; for example, chaining a database of approximately 30,000 newspaper articles (about 85 MB of text) takes 5 hours, compared to 15 minutes for a traditional IR system.

If we want to process text as quickly as possible, we must accept some errors, or at least some bad decisions. For example, consider the two portions of text shown in Fig. 1. The words *kid* and *speaker* are in the same chain, because a *speaker* can be a kind of human, as is a *kid*. This is clearly the incorrect sense of the word in this case. On the other hand, there are some advantages, such as the fact that many multi-word terms (e.g., *United States*) are taken whole, rather than as separate terms. Another benefit is the sense disambiguation mentioned above, which will help us to handle the problem of polysemy.

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 AT&T and **Residence**[2] **Inns**[7] in the **United States**[6], for **example**[3], suggests that **business**[3] **travelers**[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] **travelers**[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.

Fig. 1. Two portions of a text tagged with chain numbers.

---

3. Building links within a document

Morris and Hirst (1991) demonstrated that the structure of the lexical chains in a document corresponds to the structure of the document itself. If the lexical chains do indicate the structure of the document, then they are a natural tool to use when attempting to construct a hypertext representation of a document. In order to build these *intra-document* links, we need to determine the mapping between the lexical chains and the parts of a document's structure. We will choose the paragraph as the basic unit of a document, since this can usually be determined without any explicit document mark-up.

3.1. Determining the importance of a chain

The first step in making this determination is to decide how "important" each chain is to each paragraph in a document. By making this calculation, we will be able to link together paragraphs that share sets of important chains. We judge the importance of a chain to a particular paragraph by calculating the fraction of the content words (i.e., those words that are not stop words) in 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:

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

where $w_{c,p}$ is the number of words from chain c in paragraph p and w_p is the number of content words in paragraph p. So, if we consider the first paragraph in Fig. 1, we see that there are 9 words from chain 1 and 48 content words so $d_{1,1} = 0.19$. The result of these calculations is a set of *chain density vectors*, one for each paragraph in a document. Each of these vectors contains an element for each of the chains in the document.

3.2. Computing paragraph similarity

Once we have the set of chain density vectors, the second stage of intra-document linking is to compute the similarity between the paragraphs of the document by computing the similarity of the chain density vectors that represent them. We can compute the similarity between all pairs of paragraphs, using any one of 16 similarity coefficients. In any case, the result is a symmetric $p \times p$ matrix, where p is the number of paragraphs in the document. From this matrix we can compute the mean and the standard deviation of the paragraph similarities.

3.3. Deciding on the links

The next step is to decide which paragraphs should be linked, on the basis of the similarities calculated in the previous step. This decision can be made by looking at how the similarity of two paragraphs compares to the mean paragraph similarity across the entire document. If two paragraphs are more similar than a given threshold, then a link is placed between them. The threshold is described in terms of a z-score, that is, as a number of standard deviations from the average similarity for a particular document.

This z-score metric of similarity is meant to capture our intuition that we want to link paragraphs that are "very similar". We need such a measure because how similar two paragraphs are considered to be depends on the context in which they occur. Documents with many large chains spread throughout them will tend to display higher inter-paragraph similarity scores. If we set a simple threshold to determine which paragraphs to link, then in such cases almost all pairs of paragraphs would tend to be linked. This is clearly not the right thing to be doing, as this would severely disrupt the reader. What we would like to do is to link only those paragraphs whose similarity significantly deviates from the average. The z-score measure that we have proposed is a traditional method for determining how much a single number stands out from the mean.

4. Building links between documents

While it is interesting to build links within documents, if we have a large collection of documents that we wish to browse, then we must be able to also build links *between* documents. Our aim is to build hypertext links that will account for the fact that documents that are about the same thing will tend to use similar (although not necessarily the same) words. A link between two documents could be built by considering how the words that make up the chains contained in the document are related. By using the lexical chains extracted from the documents, rather than just the words, we can begin to account for the problems of synonymy and polysemy and we can take into account some of the more distant relations between words.

4.1. Weighted synset vectors

Each document in a database can be represented by two vectors. Each vector has an element for each synset in WordNet. An element in the first vector contains 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 contains a weight based on the number of occurrences of that particular synset when it is one WordNet relation (e.g. ANTONYMY) away from a synset that appears in the first vector. These vectors are called 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. Synsets that are the most heavily weighted in a document appear frequently in that document, but infrequently in the entire database. The weights are calculated using the standard tf·idf weighting function.

These synset weight vectors can be seen as a *conceptual* or *semantic* representation of the content of a document, as opposed to the traditional IR method of representing a document by the words that it contains. This representation also addresses both synonymy and polysemy. Synonymy is addressed by virtue of the fact that all of the synonyms of a word will be collected in the same synset, and will therefore be represented in the same element of the synset vectors. Because of the sense-disambiguation performed by the lexical chainer, a word will be represented only by synsets (i.e., senses) that are appropriate in the context of the document. Only these synsets will appear in the weighted synset vectors, solving (to some extent) the problem of polysemy. Given these two vectors, we can compute the relatedness of two documents D_1 and D_2 by computing three similarities (shown by the lines in Fig. 2):

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

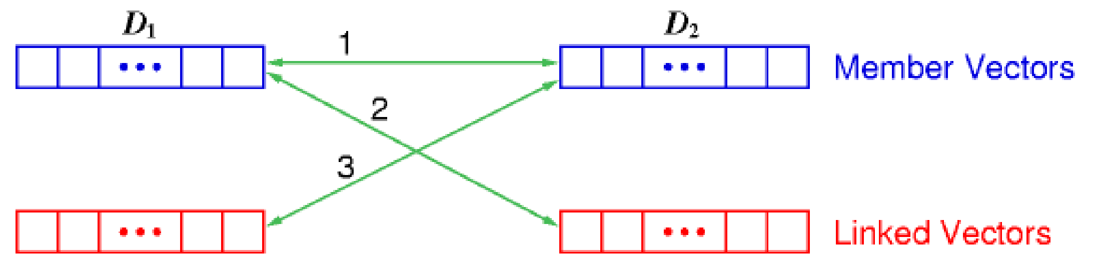


Fig. 2. Computing document similarity.

Clearly the first similarity measure is the most important since it captures not only term repetition, but also relations between documents due to synonymy. The other two relations are less important as they capture more-distant relations between synsets in WordNet. We do not calculate the linked-linked similarity, since this would allow too-distant relations between synsets in WordNet.

The process of building links between documents is relatively simple. Given a document we want to build links from, we can compute the similarity between the document's synset weight vectors and the vectors of all other documents. If the member-member similarity exceeds a set threshold, then we can calculate the two member-linked similarities and place a link between the two documents. We can rank the links using the sum of the three document similarities that we compute. Our work shows that a threshold of 0.15 will include most related documents while excluding many unrelated documents. In a sample of approximately 500,000 inter-document similarities calculated from 20 different documents, only 584 met or exceeded our threshold of 0.15.

5. Evaluating the linking methodology

There is clearly a need for evaluation when proposing a methodology such as the one described above. The question that we want to ask is: Is the methodology superior to other methodologies that have been proposed (e.g., that of Allan, 1995)? An obvious way to answer the question is to test whether the links generated by the proposed methodology will lead to better performance when they are used in the context of an appropriate IR task.

The null hypothesis for the tests is simply that there is no significant difference between the hypertext links generated by the two methodologies - one could perform IR tasks equally well using either kind of links. The alternative hypothesis is that the proposed method is superior to traditional methods because it is based on semantic relatedness, rather than strict term repetition.

5.1. The 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 using *only* hypertext browsing. This was necessary because in the interface given to the test subjects, no query engine was provided.

It may be argued that the restriction to strict hypertext browsing creates an unnatural setting for the study and that in any real system, users would at least be able to perform a keyword search. This may be true, but if we had included a query engine, then it is possible that any results that we obtained would pertain more to the use of queries as opposed to browsing or to how well users can form queries. By making the restriction, we tested just that hypothesis in which we were interested: is a semantically-based approach to hypertext link generation better than a strict term-repetition approach? If we can make a determination one way or the other, then we will be able to draw conclusions about how hypertext links should be built in a system that provides both querying and browsing.

5.2. The questions and the database

The most difficult part of performing an evaluation of any IR or hypertext system is developing reasonable questions and then determining which documents from the test database contain the answers. Several test collections have been developed over the years that can be used by anyone who wishes to compare the performance of her IR system to others. The most recent, and certainly the largest, of these collections is the TREC collection (see Harman, 1994 for a description of TREC).

From the 50 available TREC topics, three were selected to be used as the basis for the questions given to the subjects. The documents relevant to these queries (about 2,000) were extracted from the TREC databases. These were randomly combined with a random selection of other TREC documents to give a database of approximately 30,000 documents, most of which were newspaper articles.

5.3. Whose links to use?

We considered two possible methods for generating inter-document hypertext links. The first is the method described above. The second method uses an IR system called Managing Gigabytes (MG) (Witten et al., 1994) to generate links by calculating document similarity. We used the MG system to generate links in a way very similar to that presented in Allan (1995).

Links from a source document were built by passing the entire text of the source document to the MG system as a "query". MG builds the term vector representing this query after removing stop words and stemming the words in the query. This query vector was compared against the document vectors stored in the MG database, and the top 150 related documents were returned and used as the targets of the inter-document hypertext links. The MG system provided most of the same capabilities as the SMART system used by Allan. We used the MG system because it was much more easily integrated into our other software. 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*.

At this point two approaches to testing the effectiveness of these two sets of links are possible. The first is to set two (or more) experimental conditions: one using HT links and the other using MG links. This is a very typical experimental strategy, and certainly viable in this case. The problem was that such a design would have required a large number of subjects to be tested in each condition to ensure that the study was valid.

The second experimental strategy, which we used for our evaluation, is to combine the sets of links generated by the two methods at each stage during a subject's browsing. This results in a single experimental condition where the system must keep track of how each inter-document link was generated. 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 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-document links were combined by simply taking the *unique* links from each set, that is, 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. This appears to be a necessary difficulty of this method and, as we shall see, the number of correct answers that the subjects found was generally quite low, but it was nonetheless sufficient to compare the two methodologies.

The intra-document links that were presented to the users were generated by the methodology described above. Because there was no other method for generating these links, the subjects were presented only with links generated by our method.

5.4. Examining the data

The number of both inter- and intra-document 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 documents 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-document 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 in the following way:

$$L_{\mathrm{R}} = \frac{L_{\mathrm{HT}}}{L_{\mathrm{MG}}}$$

If $L_{\mathrm{R}} > 1$, 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 posed. We can explore this relationship using two different regression analyses, one incorporating only inter-document links and another incorporating both inter- and intra-document links. These analyses will express the relationship between the number of links followed and the number of correct answers found.

5.5. Inter-document links

In the first case, we can consider only the relationship between the kind of inter-document links followed and the number of answers found. This can be accomplished using a multivariate regression analysis where L_{HT} and L_{MG} are the independent variables and *Ans* is the dependent variable. Thus for each subject we will have three measurements of both the independent and dependent variables, corresponding to each of the three questions.

Note that we will enforce the condition that the constant in the model must be 0, since the subjects could not find any answers if they followed no links. Such a regression model yields the following equation:

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

So, it would seem that there is some benefit from following an HT link over an MG link. Table 1 shows the 95% confidence intervals for the model coefficients. 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.

Table 1: 95% confidence intervals for a model with inter-document links only.

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

Note that there is an overlap in the 95% confidence intervals for the coefficients of L_{HT} and L_{MG}, so we cannot reject our null hypothesis in this case. By inspection we find that the confidence intervals begin to overlap around the 92.5% confidence interval.

Rather than casting our data as a three dimensional regression problem, we could instead consider the relationship of our ratio measure, L_R to the number of answers found. If we can show that the regression line in such a model has a positive slope, then we will know that increasing the number of HT links that a user takes will increase his or her number of correct answers.

This model gives us the following equation:

$$Ans = 3.65 + 0.56 \cdot L_R \qquad (R^2 = 0.05)$$

Figure 3 shows a scatter plot of the data and the regression line. Note that the slope of the line is positive, indicating a greater benefit from HT links. Unfortunately, the 95% confidence interval for L_R contains negative values, so once again we cannot reject our null hypothesis.

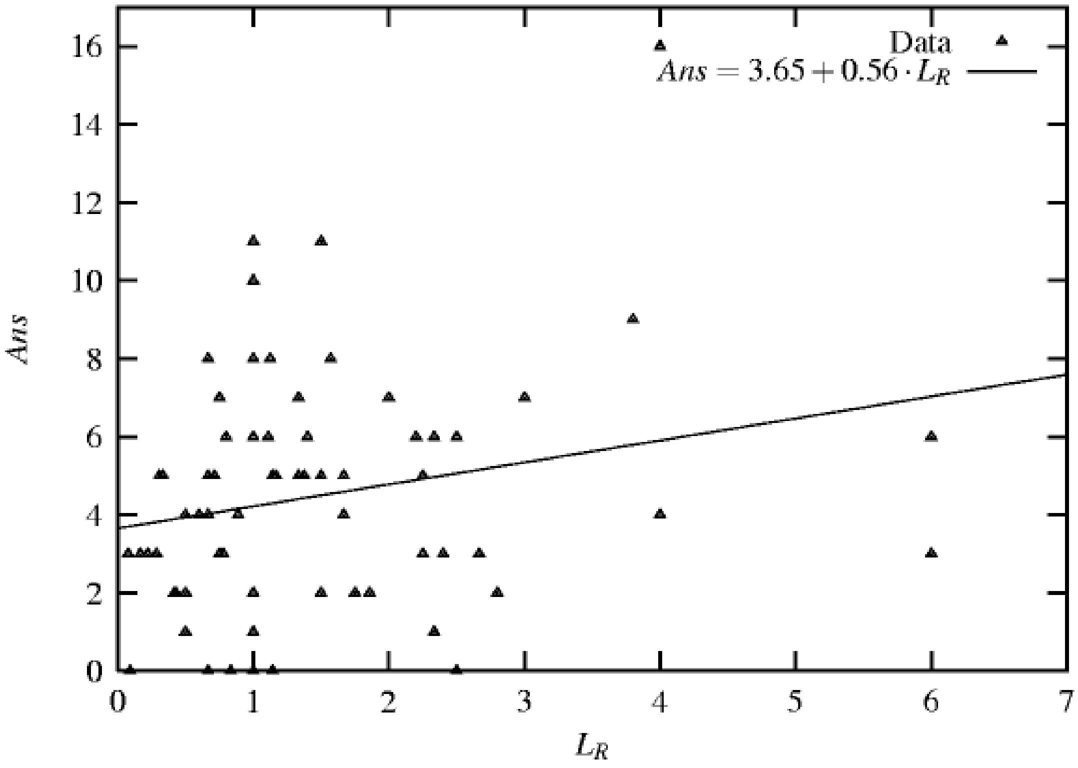


Fig. 3. Data and regression line for all subjects.

5.6. Inter- and intra-document links

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

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

Once again we see that there is an advantage in following an HT link over an MG link. Notice that, according to this model, that there is a small benefit from following an intra-document link. Unfortunately, the probability that the coefficient of L_I is 0 is unacceptably high ($p > 0.18$). Note also that the 95% confidence intervals for the coefficients of L_{HT} and L_{MG} overlap by a small margin. Thus we cannot say that intra-document links have any overall significant effect on the performance of users and we are still unable to reject the null hypothesis.

5.7. Data by experience

Studies have shown (see, for example, Marchionini, 1993) that novice users of an information system will tend to use browsing as a major component of their search strategies. Given that this is the case, we can divide our subjects into two groups on the basis of their experience using hypertext information systems (in this case the World Wide Web) and compare their performance. The two groups are as follows:

High Web Group

Subjects who use the Web three or more times a week.

Low Web Group

Subjects who use the Web less than three times a week.

Unpaired *t*-tests show that the High Web group followed significantly more inter-document links than the Low Web group, which would seem to indicate that the High Web group were more comfortable in a hypertext environment. In fact, this significant difference is due entirely to the High Web group following significantly more HT links, since there is no significant difference in the number of MG links followed between the two groups. In addition, the High Web group also found significantly more answers than the Low Web group.

The Low Web group, on the other hand, followed significantly more intra-document links, indicating that they were browsing using the hypertext links, rather than skimming documents using the scroll bars. Figures 4 and 5 show the data and regression lines for the High and Low Web groups. Note that in both cases, the models show positive slopes for the graph. Unfortunately, only the model for the Low Web group is significant.

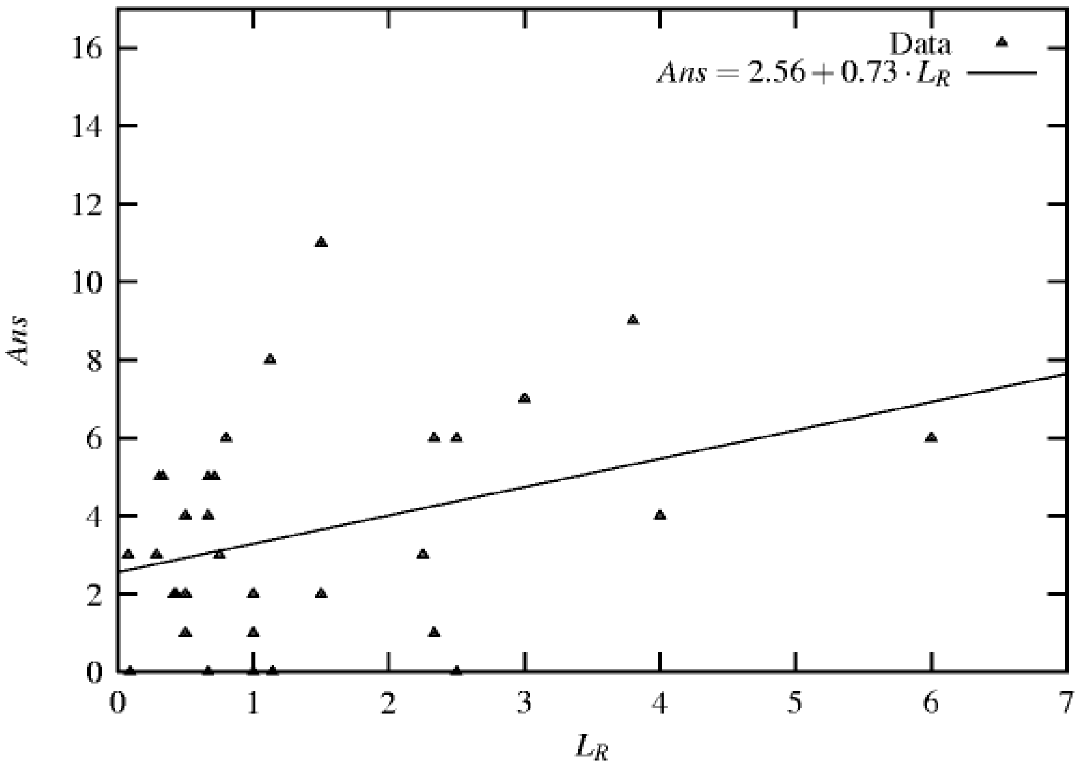


Fig. 4. Data and regression line for the Low Web group.

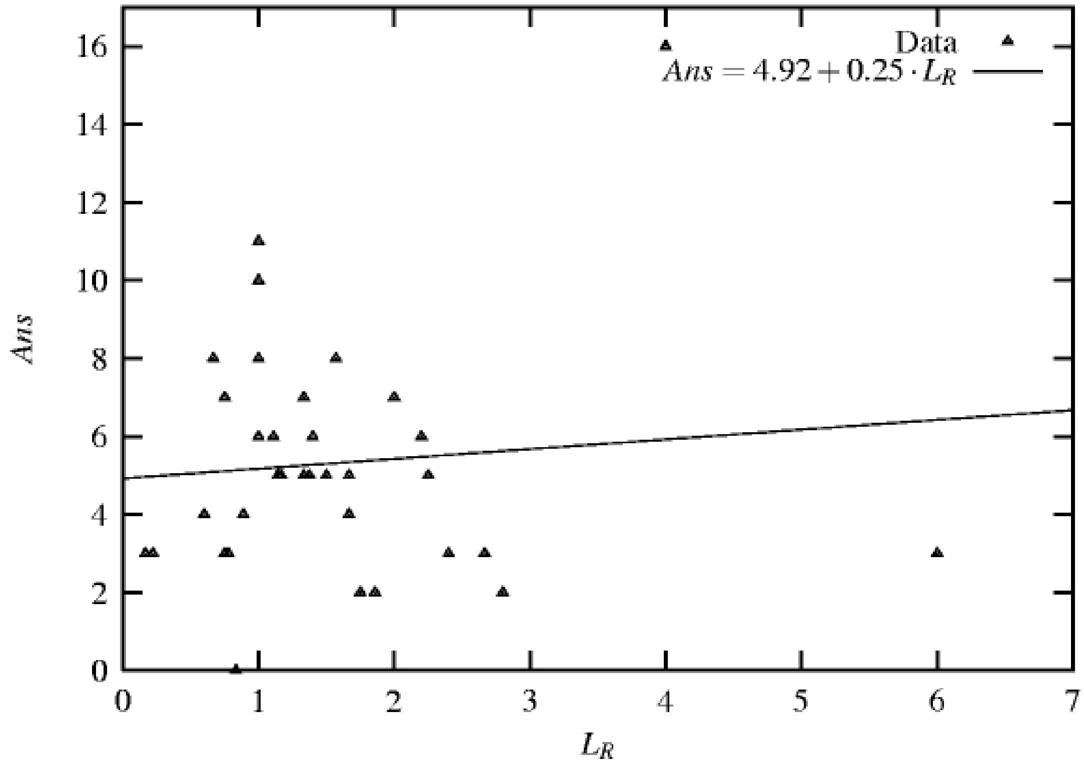


Fig. 5. Data and regression line for the High Web group.

Our two groups do, however, demonstrate that there is at least one partition of the subjects such that the only significant differences between the groups are the number of HT links followed, the number of intra-document links followed, and the number of answers found.

5.8. Viewed answers

During the course of the evaluation, subjects were limited to 15 minutes for each of their searches, which may have affected the number of answers that they could find. Because of this time limit, we decided to see how many answers were in documents that the subjects visited, even if they were not written down. We call these the *viewed answers*, and indicate the number of viewed answers that a subject encountered with the variable Ans_V.

A paired *t*-test shows a significant difference in the values of *Ans* and Ans_V. When using Ans_V as our dependent variable in a regression analysis for all subjects with all three link types, we are still unable to conclude that there is a significant advantage to using HT links over MG links.

For the High Web group, a model is produced in which there is still a high probability that the coefficient of L_I is 0. The Low Web group gives the following model:

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

The 95% confidence intervals for this model are shown in Table 2. We can see that the interval for the coefficient of L_I is always positive, indicating that the intra-document links may be of some benefit to novice users.

Table 2: 95% confidence intervals for a model with inter-document links only.

Parameter	**Value**	***t***	***p***	**High**	**Low**
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	0.4

6. Conclusions

The methodology for the automatic construction of hypertext links presented in this paper is a relatively novel one that is based entirely on semantic relatedness rather than just strict term repetition.

Given the results of the evaluation that was conducted, the answer to the question posed in the title would have to be "maybe not". Although there was no significant difference in the effect of HT links versus MG links, it seems that the difference was large enough that the methodology requires more investigation. This is motivated by the fact that there was at least one partition of the subjects (the High and Low Web groups) such that the only significant differences between them were the number of HT links followed, the number of intra-document links followed and the number of answers found. Furthermore, there was evidence that intra-document links may be of aid to novice users of a hypertext environment like the Web.

In addition to this, there were insufficiencies in some of the underlying systems used for the evaluation, most notably the lexical chainer. For example, proper nouns were completely ignored during processing, although they could be very useful. A new version of the chainer is under development and we are planning a larger evaluation.

References

J. Allan, Automatic hypertext construction. Ph.D. thesis, Cornell University, 1995.

R. Beckwith, C. Fellbaum, D. Gross, and G.A. Miller, WordNet: A lexical database organized on psycholinguistic principles, in: Uri Zernik (Ed.), *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon,*. Lawrence Erlbaum Associates, 1991, pp. 211-231.

R.L. Chapman (Ed.), *Roget's International Thesaurus*. Harper Collins, 5th edition, 1992.

D. Ellis, J. Furner-Hines, and P. 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.

S. Green, Automatically generating hypertext by computing semantic similarity. Ph.D. thesis, University of Toronto (published as Technical Report CSRG-366, 1997).

D. Harman, Overview of the 3rd Text Retrieval Conference (TREC-3), in: *Proc. of the 3rd Text Retrieval Conference*, November, 1994.

G. Marchionini, S. Dwiggins, A. Katz, and X. Lin, Information seeking in full-text end-user-oriented search systems: The roles of domain and search expertise, *Library and Information Science Research*, 15(1): 35-69, 1993.

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

B. Nordhausen, M.H. Chignell, and J. Waterworth, The missing link? Comparison of manual and automated linking in hypertext engineering, in: *Proc. of the Human Factors Society 35th Annual Meeting*, 1991.

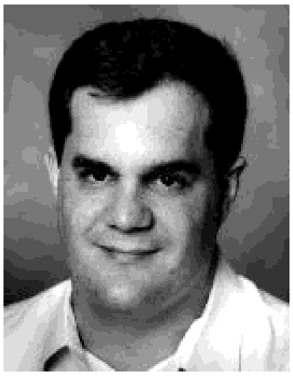
D. St-Onge, Detecting and correcting malapropisms with lexical chains. Master's thesis, University of Toronto (published as Technical Report CSRI-319, 1995).

E.M. Voorhees, Query expansion using lexical-semantic relations, in: *Proc. of SIGIR 94*, ACM, 1994

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

I.H. Witten, A. Moffat, and T.C. Bell, *Managing Gigabytes: Compressing and Indexing Documents and Images*. Van Nostrand Reinhold, 1994.

Vitae



Stephen J. Green received his Bachelor of Mathematics degree (with honours) from the University of Waterloo in 1990. His Master's work was also undertaken at Waterloo under the supervision of Professor Chrysanne DiMarco. His thesis, which was completed in 1992, was in the area of Natural Language Generation.

He subsequently began his Ph.D. studies at the University of Toronto under the supervision of Professor Graeme Hirst. His thesis work involved the automatic generation of hypertext links in large document collections, in particular, large collections of newspaper articles. He completed his Ph.D. in September, 1997 and is currently a Research Fellow in the Microsoft Research Institute at Macquarie University in Sydney, Australia.