Using lexical chains to build hypertext links in newspaper articles

Stephen J. Green
Department of Computer Science, University of Toronto
Toronto, Ontario
CANADA M5S 3G4
sjgreen@cs.utoronto.ca

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 comparing the chains contained in two different articles, we can decide whether or not to place a link between them. We also present the results of an experiment designed to measure inter-linker consistency in the manual construction of hypertext links between the paragraphs of newspaper articles. The results show that inter-linker consistency is low, but better than that obtained in a previous experiment.

Introduction

The popularity of graphical interfaces to the World Wide Web (WWW) has shown that a hypertext interface can make what was once a daunting task, accessing information across the Internet, considerably easier for the novice user. Marchionini (1989) has argued that browsing places fewer demands on the novice user, and Marchionini *et al.* (1993) report tests performed with search experts and domain experts that showed that browsing was an important component of the search technique of users unfamiliar with an information retrieval system.

Along with — and perhaps because of — the growth of the WWW, many newspapers are beginning to take their first steps into the online world. One survey puts the number of newspapers with some sort of online presence at over 100 daily newspapers[1]. The problem is that these papers are not making full use of the hypertext capabilities of the WWW. In general, the hypertexts offered are shallow; the user might find a particular article from a particular issue using hypertext links, but they must then read the entire article to find the information that interests them. It would be more useful (especially to the novice user) if hypertext links were available within and between the articles.

Westland (1991) has pointed out the economic constraints in building large-scale hypertexts. Clearly, manual construction of large-scale hypertexts from newspaper articles would be an expensive and time-consuming task, given the volume of newspaper and newswire articles produced every day. This could certainly account for the state of current WWW newspaper efforts. Bernstein (1990) has designed a hypertext apprentice that discovers possible links and alerts a human linker to them, but his system was designed to aid in the construction of hypertexts from a single large document, rather than from a large collections of documents.

Previous research on the automatic construction of hypertext (Allan 1995; Chignell *et al.* 1990; Bernstein 1990) has focused on the use of term repetition to determine what parts of a document (or documents) are related. While this method has shown promise, it is susceptible to the problem of word-sense ambiguity, and may not work for shorter texts where there will not be enough term repetition for term-weighting schemes to function. With this in mind, we are currently working on a method for automatically constructing hypertext links within and between newspaper articles using *lexical chains* (Morris & Hirst 1991).

Testing inter-linker consistency

When automatically constructing hypertext, we need to know when we have generated a "good" hypertext. The obvious choice is to take manually linked hypertexts and assume that they are "good", and then train our algorithm to produce similar hypertexts. There are two problems with this approach. Firstly,

[1] This figure is taken from NewsLink: http://www.newslink.org.

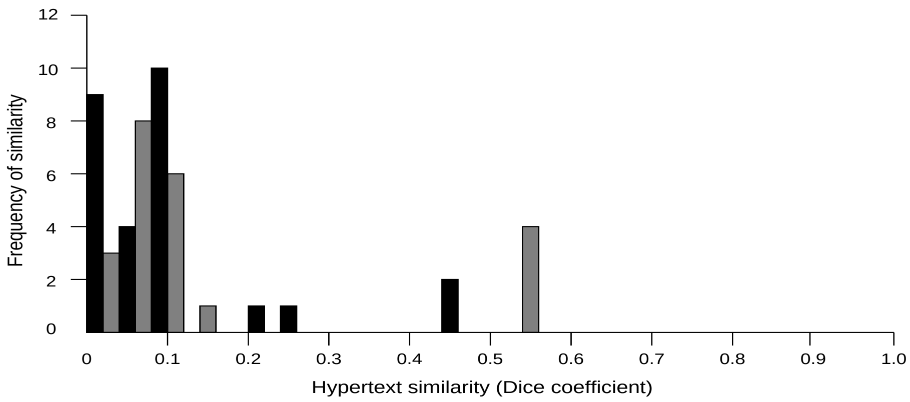

Figure 1: Histogram of similarity frequencies for technical documents, from Ellis *et al.* (1994).

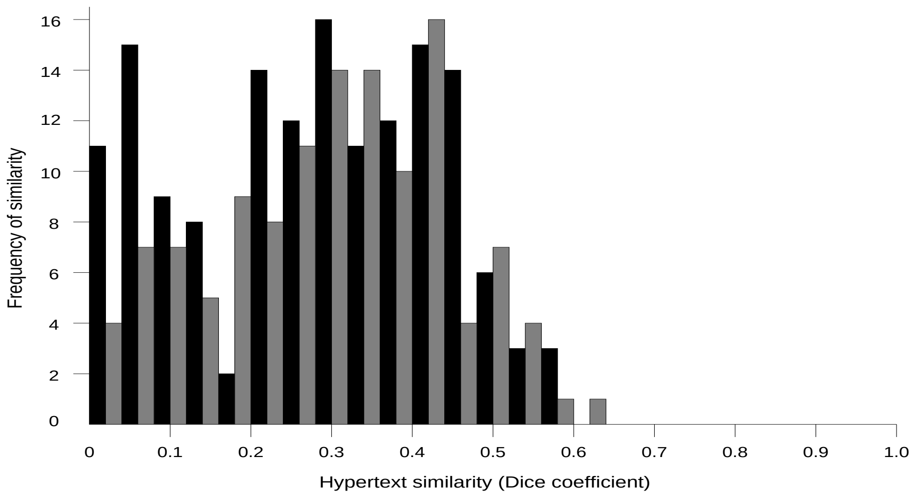

Figure 2: Histogram of similarity frequencies for newspaper articles, from our replication.

it is difficult to find a large number of manually-linked documents, since this process is extremely time-consuming. Secondly, using such hypertexts raises the question of how good human linkers are. One way to measure how humans perform at this task is to measure the consistency of several people manually linking the same document. In this case, consistency is measured by calculating the similarity between different hypertext versions of the same text.

This is the approach of Ellis *et al.* (1994a; 1994b). Their hypothesis was that the process of assigning hypertext links in a document would be similar to the process of assigning index terms to documents, a task at which humans are notoriously inconsistent. This inconsistency in assigning index terms has been shown to reduce the effectiveness of information retrieval systems that rely on these index terms.

In order to demonstrate that humans are inconsistent linkers, they had five subjects manually assign links in each of five technical documents (Ph.D. theses and journal articles). The similarity between hypertext versions was calculated using the graph representations of the hypertexts and several similarity coefficients from the Information Retrieval literature. Inter-linker consistency would be indicated by a high frequency of highly similar hypertext pairs.

As Ellis *et al.* expected, they found that the similarities between hypertext versions of the same document were very low and variable, indicating that inter-linker consistency was low. In their tests, the mean similarity across 50 document pairs (using the Dice coefficient of similarity[2]) was 0.116. The 95% confidence interval

[2] The Dice coefficient for two vectors with n elements is: $(2\sum(x_i \cdot y_i))/(\sum x_i^2 + \sum y_i^2)$, for $1 \leq i \leq n$. This is an association coefficient whose value may range between 0.0 and 1.0

for this mean was (0.069, 0.163). Figure 1 shows the frequency histogram for the similarities calculated between all 50 possible hypertext pairs. Notice that the graph is heavily skewed towards 0, indicating a high frequency of *low* similarity measures.

Ellis *et al.* suggested that the experiment be repeated with shorter documents. Due to the time-consuming nature of the task, the number of hypertexts that they were able to collect was small (five hypertext versions of each of five documents, allowing only 50 hypertext pairs). The nature of the documents linked (i.e., their length and complexity) may have also had an adverse effect on inter-linker consistency. This raises the question of how linkers would fare when presented with shorter documents that have undergone a strict editorial process and are written with a more regular structure.

We therefore replicated this experiment, using three different newspaper articles that were linked by 14 subjects (giving a total of 273 hypertext pairs). Our results indicate that the consistency was higher for the newspaper articles. Figure 2 shows the frequency histogram for the similarities that were calculated from our hypertexts, also using the Dice coefficient of similarity. In contrast to Ellis *et al.* 's results, the mean similarity across 273 document pairs in our test was 0.285, which is well outside of the confidence interval for the mean of the first experiment. An unpaired t-test indicates that the difference in the means for the two experiments is significant at the $p \leq 0.01$ level.

Unfortunately, the inter-linker consistency that we observed was still low — too low to consider using the hand-linked articles to train our algorithm. Even if the consistency had been higher, the production of a large number of articles would be entirely too costly, even considering the fact that the articles are much shorter. Still, the hand-linked articles provide useful insights into how and why paragraphs should be related.

It is possible that the reason that humans have difficulty linking paragraphs in newspaper articles is that, even though there are different aspects to the story, an article is generally about a single thing. This surface-level similarity of the aspects of a news story may be seen as “noise” that can distract the linkers, causing them to either place links between unrelated paragraphs or not place links between related paragraphs. A possible followup experiment would determine how humans would place links *between* articles. In this case, it seems that the decisions as to which articles are related would be more straightforward, and we would therefore expect to see a greater consistency between linkers.

Lexical chains

Lexical chains (Morris & Hirst 1991) are sequences of semantically related words that occur throughout a text. Generally speaking, a document will contain many chains, each of which captures a portion of the cohesive structure of the document. For example, the words *apple* and *fruit* could appear in a chain together, since *apple* is a type of *fruit*. The chains contained in a text will tend to delineate the parts of the text that are “about” the same thing. Morris and Hirst (1991) showed that the organization of the lexical chains contained in a document mirrors, in some sense, the discourse structure of that document.

The chains can be built using any lexical resource that relates words semantically. While the original work was done using *Roget's Thesaurus* (Chapman 1992), our current chainer (St-Onge 1995) uses the WordNet database (Beckwith *et al.* 1991). 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 also correspond to specialization, but in a very specific way. For example, the antonymy relation in WordNet is given a direction of horizontal, since it specializes a word very accurately.

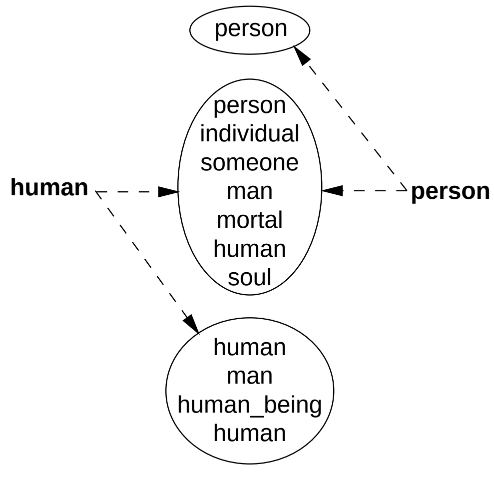

Figure 3: A strong relation between words that share a synset.

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

Extra strong Extra strong relations exist between repetitions of the same word.

Strong Strong relations exist between words that are in the same WordNet synset (i.e., words that are synonymous), as in figure 3. Strong relations also exist between words that have synsets connected by a single horizontal link (as in figure 4), or words that have synsets connected by a single IS-A or INCLUDES relation (as in figure 5).

Regular A regular relation exists 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 shorter than a given (small) length 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.

Figure 6 shows the regular relation that can be built between *apple* and *carrot*.

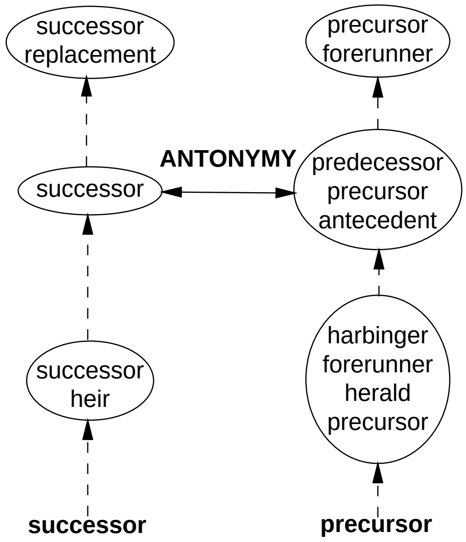

Figure 4: A strong relation between words that are antonyms.

The result of lexical chaining is a version of the text where each word is tagged with its chain number. An example of this is shown in figure 7, which shows the first and fifth paragraphs of a news article about Toronto's Police Services Board. Here, chain numbers are indicated with superscripts. A useful side effect of lexical chaining is that words are progressively sense-disambiguated as the chaining process proceeds.

The current implementation of the lexical chaining algorithm has a few drawbacks. Currently, words that

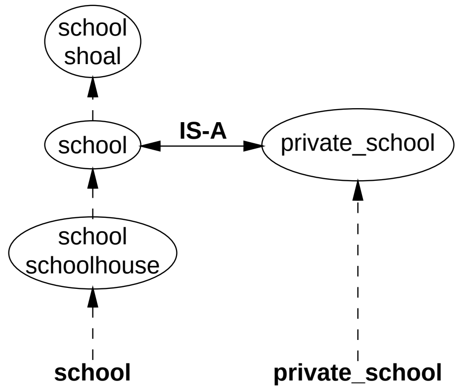

Figure 5: A strong relation between words connected by an IS-A relation.

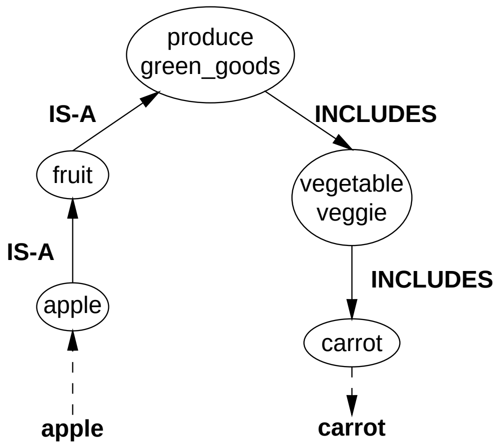

Figure 6: A regular relation connecting *apple* and *carrot*.

do not appear in WordNet are not included in lexical chains, even if they are repeated, so useful information (e.g., a chain containing all instances of a proper noun) is lost. The chainer is also relatively slow, which causes problems when one wishes to process large (in the tens of megabytes) volumes of text. Another problem, not related to the implementation, is that the WordNet database is relatively unconnected, that is, it is difficult to capture relations between nouns and verbs, since the noun and verb hierarchies are connected only at the top level.

Linking paragraphs

We can use the distribution of the lexical chains in an article to build links between the paragraphs. We do this by computing the *density* of each chain in each paragraph. The density of chain c in paragraph p, d_{cp}, The police[1] association[1]'s call[2] for Susan Eng's ouster[3] as head[2] of Metro[4]'s police[1] board[5] is being called an attempt[6] to use a personality[1] conflict[1] to disguise[6] the union[1]'s failure[1] to adequately explain new regulations[6] the officers[1] don't like.

The Metropolitan Toronto[7] Police[1] Association[1] has accused[1] Ms. Eng of conspiring with some police[1] board members[1] to sanction[10] the chief[1], and possibly even to fire[5] him, over his handling[8] of the police[1] job[6] action[1] — an allegation[12] Ms. Eng vehemently denies but which Chief[1] McCormack has not bothered to correct.

Figure 7: Two portions of a text tagged with chain numbers.

is defined as:

$$d_{cp} = \frac{w_{cp}}{w_p}$$

where w_{cp} is the number of words from chain c that appear in paragraph p and w_p is the number of content words in p. Content words are simply all those words that are not stop words. Our stop word list currently consists of 226 words, taken from the WordNet distribution, most of which are closed-class or high-frequency words. This density is computed for each chain in each paragraph. The result is that each paragraph has associated with it a vector of chain densities, one for each lexical chain.

The similarity between two paragraphs in an article can then be computed by computing the similarity between the chain density vectors associated with them. This similarity can be calculated using any one of 16 similarity coefficients that we have taken from Ellis *et al.* (1994a). These 16 similarity coefficients include both distance coefficients (where smaller numbers indicate a greater similarity) and association coefficients (where greater numbers indicate a greater similarity).

Although the similarity between paragraphs can be calculated using the chain density vectors as they are computed from the paragraphs of the article, this does not take into account Morris and Hirst's intuition that some chains are more important (or stronger) than others. Thus, the chain density vectors can be weighted using one of three different weighing functions:

Stairmand weighting This is a weighting function due to Stairmand (1994) that computes a weight for each chain in a document by considering the distance between successive paragraphs that contain elements of the chain. This function will increase[3] the density for those chains that have many elements that occur close together.

Chain length Each element of the chain density vector is weighted by considering the total length of the chain, that is, the total number of elements in the chain (including term repetitions). By using this function, we will increase the density of each chain depending on the number of elements in the chain, the intuition being that long chains represent major aspects of an article, and so they should contribute more towards the decision to link two paragraphs.

Overall density Each element of the chain density vector is weighted by considering the density of that chain throughout the entire article (i.e., the number of elements of the chain divided by the total number of content words in the document.) This function increases the density for chains that are long with respect to the length of the document, i.e., this is a measurement of relative chain length.

In addition, the vectors can be normalized to either a unit length or a zero mean.

Once we have the set of (possibly weighted and normalized) chain density vectors, the second stage of paragraph linking is to compute a similarity matrix for the story. Each element of the matrix corresponds to the value of the similarity function calculated for two chain density vectors. The result of this stage is a symmetric $n \times n$ matrix (where n is the number of paragraphs in the article). Using this matrix we can calculate the mean and the standard deviation of the paragraph similarities. Given these statistics, we can convert each similarity into a z-score. If two paragraphs are more similar than a given threshold (given in terms of a z-score) then they can be linked. The result is a (symmetric) adjacency matrix showing the links between paragraphs. This adjacency matrix is used to render an HTML version of the hypertext for display with any WWW browser.

[3] Note that we are using the term "increase" only for simplicity's sake. Whether the weighting function increases or decreases the density of a particular chain depends on whether we are using an association coefficient or a distance coefficient, respectively, to calculate the similarity between the density vectors.

The police^{1} association^{1}'s call^{2} for Susan Eng's ouster^{3} as head^{2} of Metro^{4}'s police^{1} board^{5} is being called an attempt^{6} to use a personality^{1} conflict^{1} to disguise^{6} the union^{1}'s failure^{1} to adequately explain new regulations^{6} the officers^{1} don't like.

Number of content words: 25
Chain 1: 0.320 Chain 2: 0.080 Chain 3: 0.040 Chain 4: 0.040 Chain 5: 0.040 Chain 6: 0.120

The Metropolitan Toronto^{7} Police^{1} Association^{1} has accused^{1} Ms. Eng of conspiring with some police^{1} board members^{1} to sanction^{10} the chief^{1}, and possibly even to fire^{5} him, over his handling^{8} of the police^{1} job^{6} action^{1} — an allegation^{12} Ms. Eng vehemently denies but which Chief^{1} McCormack has not bothered to correct.

Number of content words: 26
Chain 1: 0.346 Chain 5: 0.038 Chain 6: 0.038 Chain 7: 0.038 Chain 8: 0.038 Chain 10: 0.038
Chain 12: 0.038

Figure 8: Linking two paragraphs.

For example, figure 8 shows the two paragraphs from figure 7 with chain density information included. In the first paragraph, there are 8 words in chain 1 and 25 content words, so d_{11} is 0.32. When using the Dice coefficient with no weighting and no normalization to calculate the similarity between the chain density vectors for these two paragraphs, the result is 0.837. The average similarity for the entire article is 0.657, while the standard deviation is 0.158. If we are using a z-score threshold of 1.0, we can say that these paragraphs are related (The Dice coefficient is an association coefficient, so larger numbers indicate a greater similarity.)

Clearly, the choice of a specific set of parameters (a similarity coefficient, a weighting function, a normalization function, and a z-score threshold) will produce different sets of paragraph links. If we allow, say, 11 different z-score thresholds, then with four weighting functions[4], three normalization functions, and 16 similarity coefficients, we can generate $4 \times 3 \times 16 \times 11 = 2112$ not necessarily distinct hypertexts. Our current work is focused on reducing this space of possible hypertexts to a set of *representative* hypertexts. Using a method similar to the one in the experiment described above, we can compute the pairwise similarity between all of the hypertext versions of an article. If we do this for a reasonable number of articles, we will then have a large set of hypertext pair similarities, where each hypertext in a pair was generated using a different set of parameters.

The next step is to cluster these sets of parameters into five or six groups. The result would be that if we choose two sets of parameters from the same group, the hypertexts generated using those two sets would be highly similar, while two sets of parameters selected from different groups would produce dissimilar hypertexts. We could then choose a single set of parameters from each group and then use these representative sets of parameters to generate hypertexts from a large number of articles. We also hope to do some sort of analysis of variance in order to determine which of the parameters has the greatest effect on the structure of the hypertexts produced.

Linking 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. For each document that is chained, the lexical chainer outputs all of the chains found in the document. Given this kind of output, we can place a link between two articles by determining how links could be built between the chains contained in the two articles. In essence, this would be a kind of cross-document chaining.

When chaining across documents, we would restrict the chaining algorithm so that only extra strong and strong relations are allowed. We enforce such a restriction because allowing regular relations would introduce too many spurious connections. This will also ensure that building the chains across documents will be much faster than building them within documents, since we will avoid the cost of path-finding in WordNet. Along with the restriction on the types of relations between words, we will need to ensure that there is a certain minimum number of links between the chains before we can say that they are related.

We require multiple connections so that word sense ambiguity does not lead us to place a link where there

[4] The three functions described earlier and *no* weighting function.

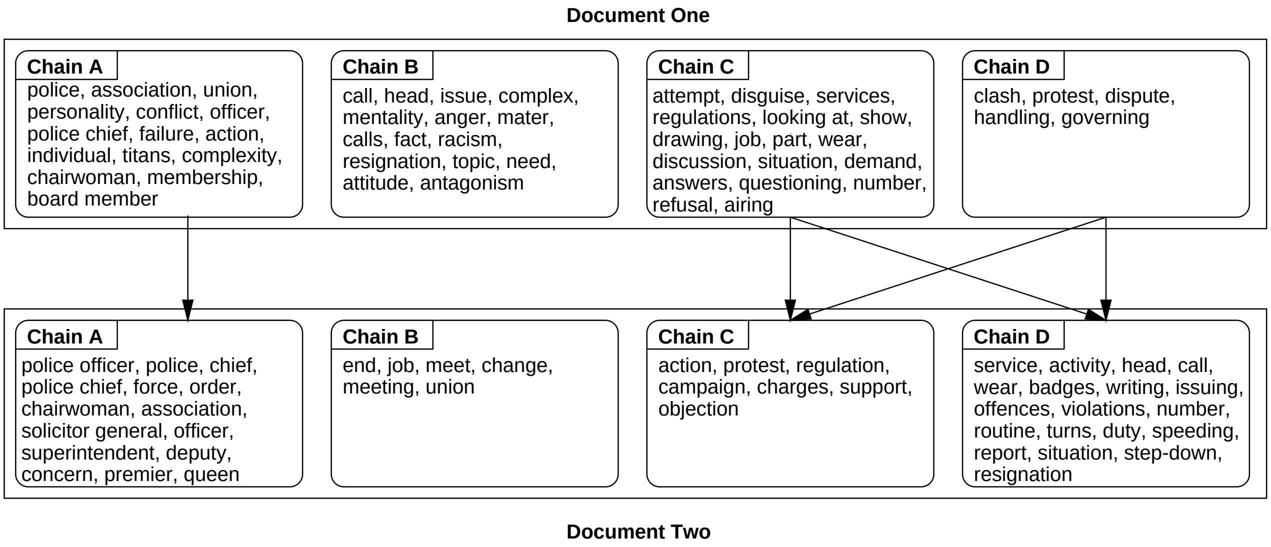

Figure 9: Links between the chains of two documents.

should not be one. Consider the following case: suppose that we allow two chains from two different documents to be related on the strength of only one link. It is possible that two chains that contain the word *bank*, for example, could be related, even though one chain uses bank in the "financial" sense, and one uses it in the "river" sense. Furthermore, consider the case where we have the word *union* in two different articles. Even if both articles use the word in the "labour movement" sense, one article may be about the police union, while the other is about the auto workers union. If we require multiple connections, then we avoid this problem, because the probability that multiple words are co-ambiguous is quite small.

This approach is similar to the local/global criteria for document similarity proposed by Salton *et al.* (1993) for use in passage retrieval systems, although the disambiguation is a natural side-effect of the lexical chaining process.

Consider the illustration in figure 9. Here we see a portion of the chains contained in two different documents that are part of a continuing story about the Toronto Police Services Board. Clearly, chain A from document one and chain A from document two are related. The relations between the other chains are less obvious. Although there is some term repetition (e.g. *protest* appears in chain D of document one and chain C of document two), it accounts for only a few of the connections between the chains.

Of course, this method could easily be used to link articles from different papers. Because we are looking for semantic relatedness, rather than strict term repetition, this method would be better able to cope with a change in language usage across different newspapers.

Evaluation

Clearly, there is a need for evaluation when building systems such as the one that we are proposing, and so we intend to perform a large-scale evaluation of our hypertext generation methodology. The evaluation will take the form of a question-answering task that will be performed over the WWW. We choose the WWW because this is where current on-line newspaper efforts are taking place and because it provides a system that potential subjects are familiar with. The database for these tests will be a large volume of newspaper articles, on the order of an entire year of one newspaper.

We have decided to use a question answering task because this is the type of task that is best done using the browsing methodology that hypertext embodies. We explicitly make no claims that our hypertext would be useful for all information access tasks, as this is clearly not the case. Our evaluation system will require a standard IR system (such as SMART) to retrieve articles to be used as starting points for browsing.

The evaluation will be designed to elicit information about the various representative hypertexts that we can generate. That is, we hope to determine which set of parameters is most useful for question answering tasks. In this case, a "good" hypertext is one that supports browsing for question answering. This question answering task will also provide information on the usefulness of intra- versus inter-article links.

We also hope to gather information about the surface characteristics of the hypertexts that we generate. Namely, we hope to determine whether subjects perform better when the intra-article link anchors are placed in the text of a paragraph, as opposed to the end of the paragraph. Similarly, we hope to determine whether inter-article links are better placed in the text or at the end of the articles.

Each subject will be provided with two questions to answer from a small pool of questions. Here we will adopt the methodology of Rada and Murphy (1992). In their experiments on searcher behaviour they used two kinds of questions: search questions and browse questions. A search question is one whose answer is contained in a single document, while the answer to a browse question may be spread across several documents. In our evaluation, we would assign each subject a search question and a browse question. Each subject will also be assigned a set of parameters for hypertext generation, along with a link anchoring strategy. We will be able measure their performance in terms of time to complete the search, links followed in searching, and whether they retrieve the correct paragraphs to answer the queries. We hope that by providing the experiment over the WWW that we will be able to have a large number of subjects.

Obviously, we will need to compare the results of using our hypertext generation methodology to other generation methodologies, most notably, that of Allan (1995).

Conclusions and future work

There are many unanswered questions in our work. One of the most obvious is: where should the intra-article links be anchored? We are currently experimenting with placing the anchors at the end of each paragraph, but we are considering whether they would be more effective when embedded in the text.

One of the advantages of Allan's work (1995) 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 (to some extent) by using lexical chains, which collect words based on 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 the anonymous referees for their comments on earlier versions of this paper. This research was supported in part by the Natural Sciences and Engineering Research Council.

References

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

Beckwith, R.; Fellbaum, C.; Gross, D.; and Miller, G. A. 1991. WordNet: A lexical database organized on psycholinguistic principles. In Zernik, U., ed., *Lexical acquisition: Exploiting on-line resources to build a lexicon.* Lawrence Erlbaum Associates. 211–231.

Bernstein, M. 1990. An apprentice that discovers hypertext links. In Streitz, N.; Rizk, A.; and André, J., eds., *Hypertext: Concepts, systems and applications: Proceedings of the European conference on hypertext,* 212–223. Cambridge University Press.

Chapman, R. L., ed. 1992. *Roget's International Thesaurus.* HarperCollins, 5th edition.

Chignell, M. H.; Nordhausen, B.; Valdez, J. F.; and Waterworth, J. A. 1990. Project HEFTI: Hypertext Extraction From Text Incrementally. Technical report, Institute of Systems Science.

Ellis, D.; Furner-Hines, J.; and Willett, P. 1994a. The creation of hypertext linkages in full-text documents: Parts I and II. Technical Report RDD/G/142, British Library Research and Development Department.

Ellis, D.; Furner-Hines, J.; and Willett, P. 1994b. On the creation of hypertext links in full-text documents: Measurement of inter-linker consistency. *The Journal of Documentation* 50(2):67–98.

Marchionini, G.; Dwiggins, S.; Katz, A.; and Lin, X. 1993. 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.

Marchionini, G. 1989. Making the transition from print to electronic encyclopedia: Adaptation of mental models. *International journal of man-machine studies* 30(6):591–618.

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

Rada, R., and Murphy, C. 1992. Searching versus browsing in hypertext. *Hypermedia* 4(1):1–30.

Salton, G.; Allan, J.; and Buckley, C. 1993. Approaches to passage retrieval in full text information systems. In *Proceedings of sixteenth annual international ACM SIGIR conference on research and development in information retreival*, 49–58.

St-Onge, D. 1995. Detecting and correcting malapropisms with lexical chains. Master's thesis, University of Toronto. Published as technical report CSRI-319, available at:
ftp://ftp.cs.toronto.edu/pub/reports/csri/319/
319.ps.Z.

Stairmand, M. 1994. Lexical chains, WordNet and information retrieval. Condensed version of Master's Thesis.

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