On the Automatic Generation of Content Links in Hypertext*

G. Salton
C. Buckley

TR 89-993
April 1989

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*This study was supported by the National Science Foundation under grant IR 87-02735.

On the Automatic Generation of Content Links in Hypertext

Gerard Salton and Chris Buckley*

Abstract

Text structuring systems that provide links between related text portions have been widely proposed as aids for text preparation and text manipulation. In principle, it is easy to follow available links between related text portions; it is much harder, however, to put in place useful links that relate document sections with related text content.

An approach is described in this note for the automatic generation of content links based on global term and phrase matches between sentence and document texts. Tentative evaluation data are included to demonstrate the usefulness of the proposed procedures.

1. Introduction

The use of structured text organizations has been widely advocated for text preparation and text manipulation. Normally, physical links may be placed in the text to relate text components and passages exhibiting appropriately defined relationships, and these links are then used to identify text excerpts that belong together and can be processed as a unit. [1-5]

It is convenient to distinguish *content links*, designed to relate document portions with similar content, from *objective links* that are more immediately concerned with overall document structure. Objective links might be used to identify certain formal text components, such as bibliographic citations and references, text footnotes and annotations, illustrations such as figures and tables, and author or publisher information. Normally, the placement of objective text links does not pose substantial intellectual problems.

The placement of content links, on the other hand, is more complex, first because the correct identification of text content is not always possible, and second because content linking is subjective, depending on the preferences and interests of individual document users, and on the circumstances surrounding text use. Content linking can however substantially simplify the document preparation by indicating to authors and editors where similar text pas-

*Department of Computer Science, Cornell University, Ithaca, NY 14853.
This study was supported in part by the National Science Foundation under grant IR 87-02735.

sages have previously occurred, and by guiding readers interested in concentrating attention to particular subject areas. If text authors or users were to be asked to supply the content links on an *ad hoc* basis, many useful links might be missed, and others could be placed incorrectly, rendering certain kinds of text manipulations difficult or impossible. On the other hand, an automatic link placement system could eliminate obvious errors, and might produce a more easily usable structure. Some approaches to the placement of content links based on automatic text analysis methodsare examined in the remainder of this note.

2. Automatic Text Analysis

A distinction may be made between global text analysis where the gross structure of large text samples, such as paragraphs and complete documents, is considered, and the detailed analysis of small text units such as individual words and phrases. In principle, one expects that the fine structure and meaning of all individual text units must be known in order to carry out a global analysis of large text components. In practice, a gross analysis that proceeds without detailed consideration of every text component will perform adequately in many applications.

When a detailed, narrow analysis is needed, including an analysis of the structure and meaning of each phrasal unit, the current approaches may depend on the construction of semantic knowledge bases covering particular subject areas, and on the use of machine-readable dictionaries from which word definitions and descriptions can be appropriately extracted. Dictionary descriptions are usable notably to obtain certain hierarchical inclusion relations between terms, as well as some synonym specifications between related dictionary entries. [6]

In general, however, the identification of individual term meaning is a largely unsolved problem even when large, machine-readable dictionaries are usable in the analysis task. More often than not, the interpretation of term meaning depends on detailed knowledge of the subject matter, and on the context in which the individual terms are used. For this reason, it is not possible, given the current state of knowledge of the semantic properties of natural language texts, to carry out detailed semantic analyses of arbitrary text samples in unrestricted document environments.

In the hypertext application, the task consists in identifying related text portions, such as document paragraphs or sentences, and in placing links between text excerpts with related text content. In such an environment, the notion of relatedness or similarity between text samples may be sufficiently broad to make it reasonable to proceed without detailed interpretation of the meaning of all text components. Global text analysis methods are therefore considered in the remainder of this note.

Over the last few decades, viable *automatic indexing* systems have been developed that are capable of determining the importance of particular terms for text content identification, and of assigning to each text a set of important content identifiers. Given a text item D_i, a content representation may then be constructed as a set, or vector, of terms $D_i = (d_{i1}, d_{i2}, \ldots, d_{it})$, in which d_{ik} represents an importance factor, or *weight*, of term T_k assigned to document D_i. [7-11] Given two documents D_i and D_j, a similarity measure may be obtained between the two items, based on the similarities between the corresponding term vectors. Typically, the similarity might be defined as an inner vector product as follows:

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

where t is the total number of assignable content terms.

A high performance term weighting system normally takes into account the frequency with which a term is used in a particular document, the number of documents in a collection to which a term is assigned, and the document length or number of terms occurring in a document. The so-called $tf \times idf$ (term frequency times inverse document frequency) strategy assigns high term weights to text elements that occur frequently inside a particular document but relatively rarely in the collection as a whole. Terms with a large $tf \times idf$ factors are known to be important for content identification purposes. [12-14]

Global similarity computations are usable for document classification by including in a common class documents exhibiting a sufficiently large pairwise global similarity. Analogously, similarity computations between documents and queries are used in information retrieval to identify stored texts that must be retrieved in response to incoming information requests. When the number of text samples to be handled is large, a pairwise comparison between all pairs of vectors may become onerous. For example, the number of sentences included in a document collection may be too large to render a complete comparison between all sentence pairs reasonable. In such circumstances, it may be more efficient to process large text excerpts first, and later to proceed to the smaller individual text units. Thus, when determining sentence similarities, it may be reasonable to process first the complete document texts as a unit, and later to consider the individual sentence pairs only for those documents exhibiting large pairwise document similarities. Such an approach also guarantees that sentence linking will depend on the value of terms in larger text environments than single sentences.

A sentence processing system capable of determining similarities between document sentences is described in the next section.

3. Sentence Comparison Process

Substantial work has been done in attempts to determine the value, or importance of individual sentences in documents. For example, automatic text abstracting strategies have been proposed based on the specification of important text words, followed by the extraction from a text corpus of sentences containing a high concentration of important text words. [15-16] The following criteria have been used among others to determine sentence importance or value:

- the number of important terms in a sentence;
- the sum of the term weights of the important terms in a sentence;
- the distance, or number of intervening words, between the important words in a sentence;
- the concentration of important terms in a sentence, defined, for example, as the number of important words divided by the total number of words in a sentence.

In the present context, the set of *important* terms is taken to be the set of terms jointly occurring in two documents exhibiting a large global similiarity coefficient. When two sentences contain a sufficiently large number of such important terms in common, a content similarity link may be placed between them. Because of the large number of potentially available sentence pairs, the sentence linking operation is best carried out in two stages, consisting first of a comparison of complete document pairs, followed by the comparison of sentences for highly matching documents. The following process may serve for the document comparison: [8-9]

1. Consider a collection of documents, or document excerpts, and identify the individual text words and sentences.
2. Remove words contained on a special list of common function words ("and", "of", "or", "but", etc.), and reduce the remaining words to word stem form, by suffix removal.

3. Assign term weights to the remaining word stems. A typical normalized term weight d_{ik} for term k in document i is

$$d_{ik} = \frac{f_{ik} \cdot \log(n_k/N)}{\sqrt{\sum_{j=1}^{t} (f_{ik})^2 \log n_k/N)^2}} \tag{2}$$

where f_{ik} represents the frequency of occurrence of term k in the document, and n_k is the number of documents with term k in a collection of N documents.

4. Compute the similarity between document pairs in accordance with the formula of equation (1) as $sim(D_i,D_{j)} = \sum_{k=1}^{t} d_{ik}\, d_{jk}$.

A set of automatically determined weighted index terms appears in Table 1 for a document dealing with cryptography. For each term, the table shows a numerical concept number assigned to that term, a term weight, and the corresponding word stem originally extracted from the document. A pairwise comparison of document vectors such as that shown in Table 1 produces global document similarity measures as shown in Table 2 in decreasing document similarity order.

In principle, the sentences contained in the highly-matching document pairs could be indexed in the same way as the complete documents, that is, using all included word stems that do not appear on a special list of excluded terms. Since the basic aim is a comparison of sentences included in two particular documents, it appears more efficient to represent the sentences by those terms only that contribute substantially to the similarity between the corresponding document pairs. For this reason, an adjusted term weight may be computed for all terms jointly contained in a given document pair. The new term weight is obtained as the product of the original weights of these terms in the two respective documents. The new weighting operation is illustrated in Table 3 for the terms jointly occurring in two sample documents. Instead of using all jointly occurring terms from a given document pair for sentence identification, the new term weighting operation makes it possible to use only the most important terms for this purpose. This is illustrated in Table 4 for five sample sentences from a given document where each sentence is indexed by one or more of the five top terms previously shown in Table 3.

The output of Table 4 shows that single terms as well as term combinations (phrases) are used for sentence indexing purposes. A *phrase* is defined simply as a set of at least two significant words occurring adjacently in the sentence text. Thus, when indexing the sentence

"the *words* in a *set* of sample texts are *decomposed* into a *set* of *word fragments* in all possible ways"

using the index set of Table 3, the recognized index set includes the single terms "word", "set", "decompose" and the phrase "word fragment". For sentence indexing, the term, or phrase weight shown in column 3 of Table 4 is taken simply as the frequency of occurrence of the corresponding term or phrase in the sentence. In the experiments described later in this note, the similarity score between two sentences S_i and S_j depends on the frequency of matching index terms in the two sentences:

$$sim(S_i, S_j) = \sum_{matching\ terms} amin\ frequency\ score\ of\ matching\ single\ terms \quad (3)$$
$$+ \sum_{matching\ phrases} amin\ frequency\ score\ of\ matching\ phrases$$

In determining the sentence similarity, it is desirable to assign greater weight to the phrase matches than to individual single term matches. This is achieved by using an appropriate multiplicative constant in expression (3). Alternatively, the assumption may be made that a phrase match between phrase [ab] consists of three distinct matching steps: match term a, match term b, and verify the word order (a precedes b in both sentences). In the experiments described in the next section, a phrase match is thus considered equivalent to a match between three different single terms.

Table 5 shows a typical comparison operation between two sample sentences. In each case, the occurrence of "word fragment" generates three indexing units, including "word", "fragment", and "word fragment", and all these units are found to be present in both sentences. In the experimental output, all sentence pairs with a similarity score of at least 5 are used for hypertext linking.

4. Experimental Hypertext Linking

Two chapters of a text on *Automatic Text Processing* were used as a test corpus, consisting of about 13,000 words of text in the areas of text compression and encryption. The sample was broken down into 93 short documents defined by 2 or 3 adjacent paragraphs of text, labeled 1 to 93. Pairwise document similarity scores were computed, as described earlier, the 100 document pairs of highest similarity being used for the sentence similarity computation. The similarity scores for 21 document pairs are shown in Fig. 1. A line joining two document nodes in the figure indicates that the corresponding document similarity proved significant (among the top 100).

The graph of Fig. 1 shows that significant document relationships exist between many documents with adjacent document numbers (for example, 30, 31, 32, 34, and so on). This is not surprising since the text for documents with adjacent numbers is extracted from the same area in the book chapter. On the other hand, relationships are also evident between many documents that are widely separated in the text, such as documents 17 and 34. In the sample at hand, document 17 covers an abstract description of text compression, whereas documents 26 to 34 contain summaries of particular text compression systems.

Corresponding to each matching document pair, one or more matching sentence pairs may be detected. Two types of sentence links are distinguished: *external* links join sentences included in two different documents, whereas *internal* links may be placed within a single document to relate sentences included in the same document. The external and internal links for the document pairs shown in Fig. 1 are presented in Figs. 2 and 3, respectively. The number of significant sentence links is noted for each document pair in Fig. 2 along the line joining the corresponding document nodes. Fig. 3 shows the actual sentence numbers of the internally linked sentences. A matching score of 5 was used as a threshold in both cases, but for internal linking, the terms from a single document only were used for sentence indexing, rather than the terms from a matching document pair.

Table 6 contains an evaluation of the effectiveness of the document and sentence linking operations for the sample exhibited in Figs. 1 to 3. The table shows that all links between document pairs were significant, and that the internal linking produced nearly perfect results (precision of 95 percent). The precision of the external linking was 76 percent. Failures occurred when the vocabularies were relatively similar for two sentences, but the points made in these sentences were somewhat distinct. Table 7 shows examples of good as well as marginal sentence linking. Both sentences in Table 7(a) provide information about the use of variable-length encoding units for text compression. In Table 7(b), both sentences use terms such as "occurrence probability" and "encoding unit", but sentence 1 covers fixed-length codes, whereas sentence 2 deals with variable-length codes, such as the Huffman code.

The small-scale experiment reported in this note does not represent the final word on sentence linking, first because the sample documents are untypical in that they all cover homogeneous subject matter produced by a single author. Furthermore, the evaluation was subjective being conducted by manually considering each available sentence pair. In principle the linked structure should be used in an operational environment to judge effectiveness. The simple experiment presented here does however make clear that a global sentence linking process may produce useful text links in the vast majority of cases. With appropriate fine-tuning of the parameters controlling the linking process, it should be possible in the future to generate useful hypertext structures automatically.

References

[1] J.B. Smith and S.F. Weiss, An Overview of Hypertext, *Communications of the ACM*, 31:7, July 1988, 816-819.

[2] D.R. Raymond and F.W. Tompa, Hypertext and the New Oxford Dictionary, *Communications of the ACM*, 31:7, July 1988, 871-879.

[3] M.E. Frisse, Searching for Information in a Hypertext Medical Handbook, *Communications of the ACM*, 31:7, July 1988, 880-886.

[4] P.J. Brown, Linking and Searching within Hypertext, *Electronic Publishing*, 1:1, April 1988, 45-53.

[5] J. Conklin, Hypertext: An Introduction and Survey, *Computer*, 20:9, September 1987, 17-41.

[6] E.A. Fox, J.T. Nutter, T. Ahlswede and M. Evens, Building a Large Thesaurus for Information Retrieval, *Proceedings Second Conference on Applied Natural Language Processing*, Association for Computational Linguistics, Austin, TX, February 1988, 101-108.

[7] C.J. van Rijsbergen, *Information Retrieval*, Second Edition, Butterworths, London, 1979.

[8] G. Salton and M.J. McGill, *Introduction to Modern Information Retrieval*, McGraw Hill Book Co., New York, 1983.

[9] G. Salton, *Automatic Text Processing*, Addison-Wesley Publishing Co., Reading, MA, 1989.

[10] G. Salton, A Theory of Indexing, *Regional Conference Series in Applied Mathematics No. 18*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1975.

[11] G. Salton, A Blueprint for Automatic Indexing, *ACM SIGIR Forum*, 16:2, Fall 1981, 22-38.

[12] G. Salton and C.S. Yang, On the Specification of Term Values in Automatic Indexing, *Journal of Documentation*, 29:4, December 1973, 351-372.

[13] G. Salton, C.S. Yang and C.T. Yu, A Theory of Term Importance in Automatic Text Analysis, *Journal of the ASIS*, 26:1, January-February 1975, 33-44.

[14] G. Salton and C. Buckley, Term Weighting Approaches in Automatic Text Retrieval, *Information Processing and Management*, 24:5, 1988, 513-523.

[15] H.P. Luhn, The Automatic Creation of Literature Abstracts, *IBM Journal of Research and Development*, 2:2, April 1958, 159-165.

[16] H.P. Edmundson and R.E. Wyllys, Automatic Abstracting and Indexing--Survey and Recommendations, *Communications of the ACM*, 7:4, April 1964, 259-263.

Document Number	Concept Number	Concept Weight	Word Stem
52	310	0.03318	system
52	990	0.04730	tranform
52	999	0.25921	sign
52	1013	0.07041	norm
52	1351	0.12960	read
52	1407	0.13950	authent
51	1453	0.07041	cas
52	1478	0.34046	satis
52	1655	0.10876	restor
52	2125	0.18929	equ
52	2315	0.18158	appl
52	2797	0.13195	deciph
52	2897	0.06194	need
52	3119	0.33111	key
52	3272	0.12152	cryptosystem
52	3499	0.13950	priv
52	3511	0.36034	mes
52	3585	0.13195	pair
52	3832	0.10876	follow
52	3875	0.09079	invers
52	4103	0.20096	immater
52	4115	0.08395	ord
52	4237	0.19774	publ
52	4459	0.04333	enciph
52	4648	0.15607	receiv
52	4723	0.15225	wish
52	4779	0.20096	ap
52	4911	0.04871	produc
52	4915	0.06391	provid
52	4955	0.41417	send

Typical Automatic Indexing Product
(single term word stems)

Table 1

Document Pair		Global Similarity
52	53	0.662501
39	40	0.648946
56	57	0.631081
57	77	0.613981
17	30	0.594061
40	41	0.583577
23	24	0.576988
17	37	0.549309
39	41	0.546994
91	92	0.541512
40	43	0.540440
90	92	0.531145
31	34	0.530627
32	33	0.530337
21	22	0.528630
69	71	0.524707
65	66	0.524520
17	34	0.522904
30	34	0.519885
58	59	0.518682
41	42	0.518432
3	4	0.516225
56	77	0.515193

Typical Pairwise Document Similarity Measures

Table 2

Original Term Weight in Two Documents		New Term Weight Equal to Product of Weights	Corresponding Word Stems
0.62908	0.82395	0.5183300	frag
0.18688	0.14398	0.0269070	set
0.18675	0.14388	0.0268696	decompos
0.21035	0.10804	0.0227262	word
0.15568	0.13993	0.0217843	frequ

Indexing Units for Sentences from Documents 39, 40

Table 3

Sentence Number	Concept Numbers	Frequency	Concept (single terms and phrases)
1	1433	1.00000	word
1	3561	2.00000	frequ
1	4539	2.00000	frag
1	4706	1.00000	set
1	3562	1.00000	word frag
2	1433	1.00000	word
2	4539	2.00000	frag
2	4706	2.00000	set
2	78	2.00000	frag set
3	3561	1.00000	frequ
3	4539	4.00000	frag
3	4706	1.00000	set
3	78	1.00000	frag set
4	1433	2.00000	word
4	1580	1.00000	decompos
4	4539	1.00000	frag
5	1433	2.00000	word
5	3561	3.00000	frequ
5	4539	1.00000	frag
5	4706	1.00000	set
5	78	1.00000	frag set
5	3562	1.00000	word frag

Sentence Indexing for Document 39

Table 4

Original Text	S_1: The *words* in a *set* of sample texts are *decomposed* into a *set* of *word fragments*.	S_2: A list of *word fragments* from a typical *set* of 256 *fragments* is considered
Initial Term Set	word, set, decompose, set, word fragment, (word), (fragment)	word fragment, (word), (fragment), set, fragment
Term Set with Frequency Weight	word 2 set 2 decompose 1 word fragment 1 fragment 1	word 1 set 1 - word fragment 1 fragment 2
Total Matching Score = 4.		

Sentence Comparison

Table 5

	Document Pairs	External Sentence Links	Internal Sentence Links
Number of significant pairs	21	122	19
Number of links judged very relevant	9	56	12
Number of links judged relevant	12	37	6
Number of links judged marginal	0	29	1
Precision of linking	21/21 = 1.0	93/122 = 0.76	18/19 = 0.95

Link Evaluation

Table 6

1. Variable-*length encoding units* might be used, constructed so as to even out the occurrence frequencies of the various *encoding units*

2. Thus the possibility of applying *fixed-length codes* to variable-*length encoding units* should be given further attention

Indexing vocabulary: code, encode, fix, length, unit

a) Relevant Sentence Linking

1. When the characters to be *encoded* have even occurrence frequencies on *probabilities*, the use of fixed-length codes of length x for n *encoding units* is in fact optimal

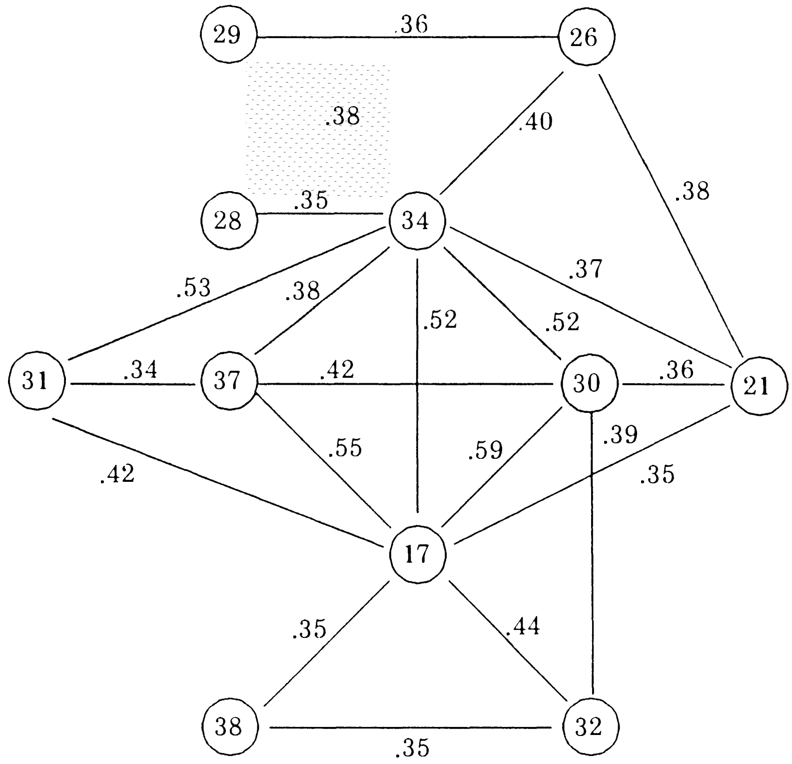
2. A Huffman *code* is generated recursively by combining two *encoding units* with the lowest *occurrence probability* while generating a new combined *encoding unit* whose *occurrence probability* is the sum of the *probabilities* of the two component *units*

Indexing vocabulary: code, encode, occurrence, probability, unit

b) Marginal Sentence Linking

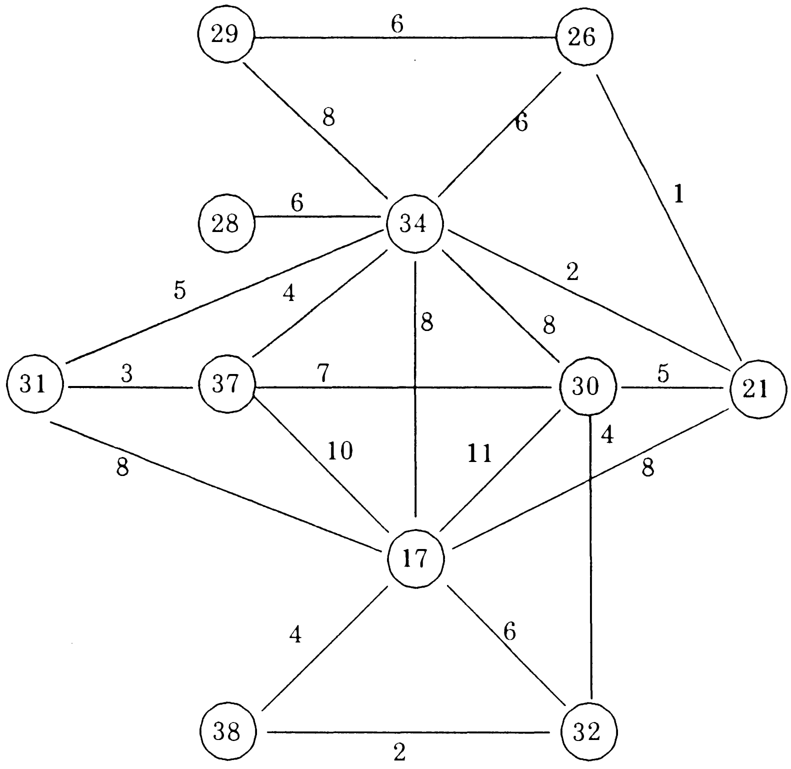
Sentence Linking Examples

Table 7



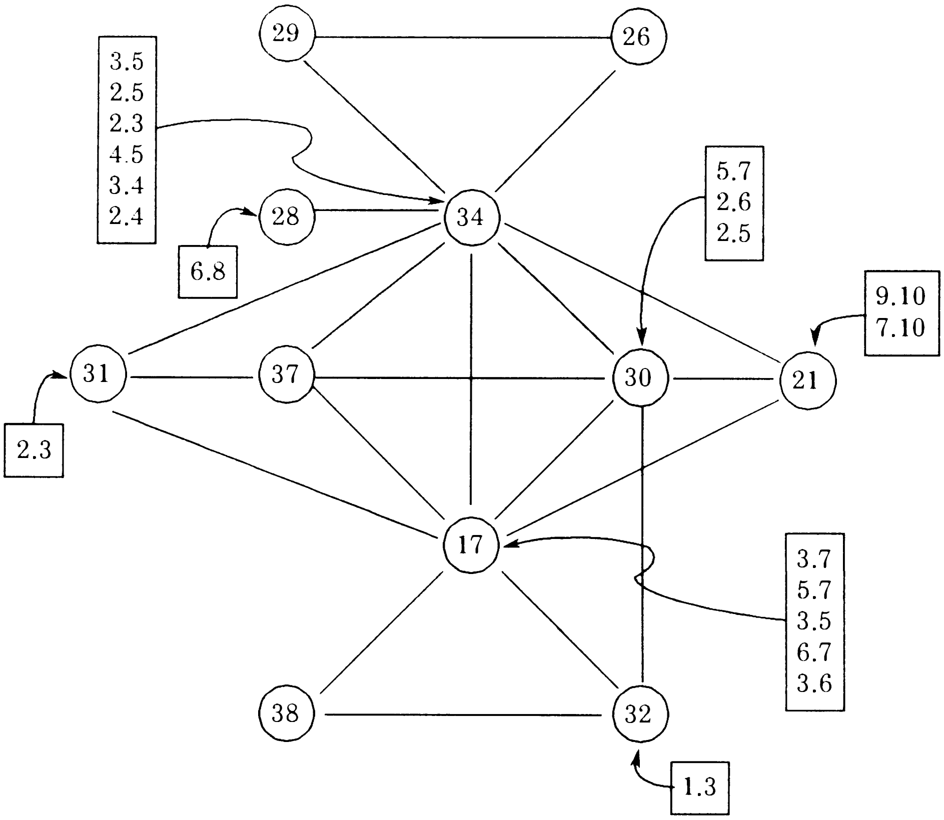
Pairwise Document Similarity Scores

Fig. 1



External Sentence Linking (total links 122)

Fig. 2



Internal Sentence Linking (total links 19)

Fig. 3