# Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text

Jane Morris*
York University

Graeme Hirst†
University of Toronto

*In text, lexical cohesion is the result of chains of related words that contribute to the continuity of lexical meaning. These lexical chains are a direct result of units of text being "about the same thing," and finding text structure involves finding units of text that are about the same thing. Hence, computing the chains is useful, since they will have a correspondence to the structure of the text. Determining the structure of text is an essential step in determining the deep meaning of the text. In this paper, a thesaurus is used as the major knowledge base for computing lexical chains. Correspondences between lexical chains and structural elements are shown to exist. Since the lexical chains are computable, and exist in non–domain-specific text, they provide a valuable indicator of text structure. The lexical chains also provide a semantic context for interpreting words, concepts, and sentences.*

## 1. Lexical Cohesion

A text or discourse is not just a set of sentences, each on some random topic. Rather, the sentences and phrases of any sensible text will each tend to be about the same things — that is, the text will have a quality of unity. This is the property of *cohesion* — the sentences "stick together" to function as a whole. Cohesion is achieved through back-reference, conjunction, and semantic word relations. Cohesion is not a guarantee of unity in text but rather a device for creating it. As aptly stated by Halliday and Hasan (1976), it is a way of getting text to "hang together as a whole." Their work on cohesion has underscored its importance as an indicator of text unity.

Lexical cohesion is the cohesion that arises from semantic relationships between words. All that is required is that there be some recognizable relation between the words.

Halliday and Hasan have provided a classification of lexical cohesion based on the type of dependency relationship that exists between words. There are five basic classes:

1. Reiteration with identity of reference:

   **Example 1**
   1. Mary bit into a *peach.*
   2. Unfortunately the *peach* wasn't ripe.

* Department of Computer Science, York University, North York, Ontario, Canada M3J 1P3
† Department of Computer Science, University of Toronto, Toronto, Ontario, Canada M5S 1A4

2. Reiteration without identity of reference:

   **Example 2**
   1. Mary ate some *peaches*.
   2. She likes *peaches* very much.

3. Reiteration by means of superordinate:

   **Example 3**
   1. Mary ate a *peach*.
   2. She likes *fruit*.

4. Systematic semantic relation (systematically classifiable):

   **Example 4**
   1. Mary likes *green* apples.
   2. She does not like *red* ones.

5. Nonsystematic semantic relation (not systematically classifiable):

   **Example 5**
   1. Mary spent three hours in the *garden* yesterday.
   2. She was *digging* potatoes.

Examples 1, 2, and 3 fall into the class of *reiteration*. Note that reiteration includes not only identity of reference or repetition of the same word, but also the use of superordinates, subordinates, and synonyms.

Examples 4 and 5 fall into the class of *collocation*, that is, semantic relationships between words that often co-occur. They can be further divided into two categories of relationship: *systematic semantic*, and *nonsystematic semantic*.

Systematic semantic relationships can be *classified* in a fairly straightforward way. This type of relation includes antonyms, members of an ordered set such as {*one, two, three*}, members of an unordered set such as {*white, black, red*}, and part-to-whole relationships like {*eyes, mouth, face*}. Example 5 is an illustration of collocation where the word relationship, {*garden, digging*}, is nonsystematic. This type of relationship is the most problematic, especially from a knowledge representation point of view. Such collocation relationships exist between words that tend to occur in similar lexical environments. Words tend to occur in similar lexical environments because they describe things that tend to occur in similar situations or contexts in the world. Hence, context-specific examples such as {*post office, service, stamps, pay, leave*} are included in the class. (This example is from Ventola (1987), who analyzed the patterns of lexical cohesion specific to the context of service encounters.) Another example of this type is {*car, lights, turning*}, taken from example 14 in Section 4.2. These words are related in the situation of driving a car, but taken out of that situation, they are not related in a systematic way. Also contained in the class of collocation are *word associations*. Examples from Postman and Keppel (1970) are {*priest, church*}, {*citizen, U.S.A.*}, and {*whistle, stop*}. Again, the exact relationship between these words can be hard to classify, but there does exist a recognizable relationship.

### 1.1 Lexical Chains

Often, lexical cohesion occurs not simply between pairs of words but over a succession of a number of nearby related words spanning a topical unit of the text. These

sequences of related words will be called *lexical chains*. There is a *distance* relation between each word in the chain, and the words co-occur within a given *span*. Lexical chains do not stop at sentence boundaries. They can connect a pair of adjacent words or range over an entire text.

Lexical chains tend to delineate portions of text that have a strong unity of meaning. Consider this example (sentences 31–33 from the long example given in Section 4.2):

**Example 6**
In front of me lay a virgin crescent cut out of pine bush. A dozen houses were going up, in various stages of construction, surrounded by hummocks of dry earth and stands of precariously tall trees nude halfway up their trunks. They were the kind of trees you might see in the mountains.

A lexical chain spanning these three sentences is {*virgin, pine, bush, trees, trunks, trees*}. Section 3 will explain how such chains are formed. Section 4 is an analysis of the correspondence between lexical chains and the structure of the text.

### 1.2 Why Lexical Cohesion Is Important

There are two major reasons why lexical cohesion is important for computational text understanding systems:

1. Lexical chains provide an easy-to-determine context to aid in the resolution of ambiguity and in the narrowing to a specific meaning of a word.
2. Lexical chains provide a clue for the determination of coherence and discourse structure, and hence the larger meaning of the text.

**1.2.1 Word Interpretation in Context.** Word meanings do not exist in isolation. Each word must be interpreted in its context. For example, in the context {*gin, alcohol, sober, drinks*}, the meaning of the noun *drinks* is narrowed down to alcoholic *drinks*. In the context {*hair, curl, comb, wave*} (Halliday and Hasan 1976), *wave* means a hair wave, not a water wave, a physics wave, or a friendly hand wave. In these examples, lexical chains can be used as a contextual aid to interpreting word meanings.

In earlier work, Hirst (1987) used a system called "Polaroid Words" to provide for intrasentential lexical disambiguation. Polaroid Words relied on a variety of cues, including syntax, selectional restrictions, case frames, and — most relevant here — a notion of semantic distance or relatedness to other words in the sentences; a sense that had such a relationship was preferred over one that didn't. Relationships were determined by marker passing along the arcs in a knowledge base. The intuition was that semantically related concepts will be physically close in the knowledge base, and can thus be found by traversing the arcs for a limited distance. But Polaroid Words looked only for possible relatedness between words in the same sentence; trying to find connections with all the words in preceding sentences was too complicated and too likely to be led astray. The idea of lexical chains, however, can address this weakness in Polaroid Words; lexical chains provide a constrained easy-to-determine representation of context for consideration of semantic distance.

**1.2.2 Cohesion and Discourse Structure.** The second major importance of lexical chains is that they provide a clue for the determination of coherence and discourse structure.

When a chunk of text forms a unit within a discourse, there is a tendency for related words to be used. It follows that if lexical chains can be determined, they will tend to indicate the structure of the text.

We will describe the application of lexical cohesion to the determination of the discourse structure that was proposed by Grosz and Sidner (1986). Grosz and Sidner propose a structure common to all discourse, which could be used along with a structurally dependent focus of attention to delineate and constrain referring expressions. In this theory there are three interacting components: *linguistic structure, intentional structure,* and *attentional state.*

Linguistic structure is the segmentation of discourse into groups of sentences, each fulfilling a distinct role in the discourse. Boundaries of segments can be fuzzy, but some factors aiding in their determination are *clue words,* changes in intonation (not helpful in written text), and changes in aspect and tense. When found, these segments indicate changes in the topics or ideas being discussed, and hence will have an effect on potential referents.

The second major component of the theory is the intentional structure. It is based on the idea that people have definite purposes for engaging in discourse. There is an overall discourse purpose, and also a discourse segment purpose for each of the segments in the linguistic structure described above. Each segment purpose specifies how the segment contributes to the overall discourse purpose. There are two structural relationships between these segments. The first is called a *dominance* relation, which occurs when the satisfaction (i.e., successful completion) of one segment's intention contributes to the satisfaction of another segment's intention. The second relation is called *satisfaction precedence,* which occurs when the satisfaction of one discourse segment purpose must occur before the satisfaction of another discourse segment purpose can occur.

The third component of this theory is the attentional state. This is a stack-based model of the set of things that attention is focused on at any given point in the discourse. It is "parasitic" on the intentional and linguistic structures, since for each discourse segment there exists a separate focus space. The dominance relations and satisfaction precedence relations determine the pushes and pops of this stack space. When a discourse segment purpose contributes to a discourse segment purpose of the immediately preceding discourse segment, the new focus space is pushed onto the stack. If the new discourse segment purpose contributes to a discourse segment purpose earlier in the discourse, focus spaces are popped off the stack until the discourse segment that the new one contributes to is on the top of the stack.

It is crucial to this theory that the linguistic segments be identified, and as stated by Grosz and Sidner, this is a problem area. This paper will show that lexical chains are a good indication of the linguistic segmentation. When a lexical chain ends, there is a tendency for a linguistic segment to end, as the lexical chains tend to indicate the topicality of segments. If a new lexical chain begins, this is an indication or clue that a new segment has begun. If an old chain is referred to again (a *chain return*), it is a strong indication that a previous segment is being returned to. We will demonstrate this in Section 4.

### 1.3 Cohesion and Coherence

The theory of *coherence relations* (Hobbs 1978; Hirst 1981; McKeown 1985) will now be considered in relation to cohesion. There has been some confusion as to the differences between the phenomena of *cohesion* and *coherence,* e.g., Reichman (1985). There is a danger of lumping the two together and losing the distinct contributions of each to the understanding of the unity of text.

Ultimately, the difference between cohesion and coherence is this: *cohesion* is a term for sticking together; it means that the text all hangs together. *Coherence* is a term for making sense; it means that there is sense in the text. Hence the term *coherence relations* refers to the relations between sentences that contribute to their making sense.

Cohesion and coherence relations may be distinguished in the following way. A coherence relation is a relation among clauses or sentences, such as *elaboration, support, cause,* or *exemplification.* There have been various attempts to classify all possible coherence relations, but there is as yet no widespread agreement. There does not exist a general computationally feasible mechanism for identifying coherence relations. In contrast, cohesion relations are relations among elements in a text: *reference, ellipsis, substitution, conjunction,* and *lexical cohesion.*

Since cohesion is well-defined, one might expect that it would be computationally easier to identify, because the identification of ellipsis, reference, substitution, conjunction, and lexical cohesion is a straightforward task for people. We will show below that *lexical* cohesion is computationally feasible to identify. In contrast, the identification of a specific coherence relation from a given set is not a straightforward task, even for people. Consider this example from Hobbs (1978):

**Example 7**
1. John can open Bill's safe.
2. He knows the combination.

Hobbs identifies the coherence relation as *elaboration.* But it could just as easily be *explanation.* This distinction depends on context, knowledge, and beliefs. For example, if you questioned John's ability to open Bill's safe, you would probably identify the relation as explanation. Otherwise you could identify it as elaboration. Here is another example:

**Example 8**
1. John bought a raincoat.
2. He went shopping yesterday on Queen Street and it rained.

The coherence relation here could be elaboration (on the buying), or explanation (of when, how, or why), or cause (he bought the raincoat because it was raining out).

The point is that the identity of coherence relations is "interpretative," whereas the identity of cohesion relations is not. At a general level, even if the precise coherence relation is not known, the relation "is about the same thing" exists if coherence exists. In the example from Hobbs above, *safe* and *combination* are lexically related, which in a general sense means they "are about the same thing in some way." In example 8, *bought* and *shopping* are lexically related, as are *raincoat* and *rained.* This shows how cohesion can be useful in identifying sentences that are coherently related.

Cohesion and coherence are independent, in that cohesion can exist in sentences that are not related coherently:

**Example 9**
Wash and core six apples. Use them to cut out the material for your new suit. They tend to add a lot to the color and texture of clothing. Actually, maybe you should use five of them instead of six, since they are quite large.

Similarly, coherence can exist without textual cohesion:

**Example 10**
I came home from work at 6:00 p.m. Dinner consisted of two chicken breasts and a bowl of rice.

Of course, most sentences that relate coherently do exhibit cohesion as well.[1]

### 1.4 The Importance of Both Cohesion and Coherence

Halliday and Hasan (1976) give two examples of lexical cohesion involving identity of reference:

**Example 11**
1. Wash and core six cooking *apples*.
2. Put *them* into a fireproof dish.

**Example 12**
1. Wash and core six cooking *apples*.
2. Put the *apples* into a fireproof dish.

Reichman (1985, p. 180) writes "It is not the use of a pronoun that gives *cohesion* to the wash-and-core-apples text. These utterances form a *coherent* piece of text not because the pronoun *them* is used but because they jointly describe a set of cooking instructions" (emphasis added). This is an example of lumping cohesion and coherence together as one phenomenon. Pronominal reference is defined as a type of **cohesion** (Halliday and Hasan 1976). Therefore the *them* in example 11 is an instance of it. The important point is that *both* cohesion and coherence are distinct phenomena creating unity in text.

Reichman also writes (1985, p. 179) "that similar words (*apples, them, apples*) appear in a given stretch of discourse is an artifact of the content of discussion." It follows that if content is related in a stretch of discourse, there will be coherence. Lexical cohesion is a computationally feasible clue to identifying a coherent stretch of text. In example 12, it is computationally trivial to get the word relationship between *apples* and *apples*, and this relation fits the definition of lexical cohesion. Surely this simple indicator of coherence is useful, since as stated above, there does not exist a computationally feasible method of identifying coherence in non–domain-specific text. Cohesion is a useful indicator of coherence regardless of whether it is used intentionally by writers to create coherence, or is a result of the coherence of text.

Hobbs (1978) sees the resolution of coreference (which is a form of cohesion) as being subsumed by the identification of coherence. He uses a formal definition of coherence relations, an extensive knowledge base of assertions and properties of objects and actions, and a mechanism that searches this knowledge source and makes simple inferences. Also, certain elements must be assumed to be coreferential.

He shows how, in example (7), an assumption of coherence allows the *combination* to be identified as the combination of *Bill's safe* and *John* and *he* to be found to be coreferential.

1 There is an interesting analogy between cohesion and syntax, and coherence and semantics. *Jabberwocky* (Carroll 1872) is an example of syntax sticking text together without semantics. Example 10 illustrates coherence sticking text together without cohesion.

But lexical cohesion would also indicate that *safe* and *combination* can be assumed to be coreferential. And more importantly, one should not be misled by chicken-and-egg questions when dealing with cohesion and coherence. Rather, one should use each where applicable. Since the lexical cohesion between *combination* and *safe* is easy to compute, we argue that it makes sense to use this information as an indicator of coherence.

## 2. The Thesaurus and Lexical Cohesion

The thesaurus was conceived by Peter Mark Roget, who described it as being the "converse" of a dictionary. A dictionary explains the meaning of words, whereas a thesaurus aids in finding the words that best express an idea or meaning. In Section 3, we will show how a thesaurus can be used to find lexical chains in text.

### 2.1 The Structure of the Thesaurus

*Roget's International Thesaurus, 4th Edition* (1977) is composed of 1042 sequentially numbered basic categories. There is a hierarchical structure both above and below this level (see Figure 1). Three structure levels are above the category level. The topmost level consists of eight major *classes* developed by Roget in 1852: abstract relations, space, physics, matter, sensation, intellect, volition, and affections. Each class is divided into (roman-numbered) *subclasses*, and under each subclass there is a (capital-letter-sequenced) *sub-subclass*. These in turn are divided into the basic categories.

Where applicable, categories are organized into *antonym pairs*. For example, category 407 is *Life*, and category 408 is *Death*.

Each category contains a series of numbered paragraphs to group closely related words. Within each paragraph, still finer groups are marked by semicolons. In addition, a semicolon group may have cross-references or pointers to other related categories or paragraphs. A paragraph contains words of only one syntactic category. The noun paragraphs are grouped at the start of a category, followed by the paragraphs for

Class 1 ···
⋮
Class 4: Matter
I ···
⋮
III Organic Matter
A ···
⋮
B Vitality
⋮
407 Life
1. NOUNS life, living, vitality, being alive, having life, animation, animate existence; liveliness, animal spirits, vivacity, spriteliness; long life, longevity; viability; lifetime 110.5; immortality 112.3; birth 167; existence 1; bio-, organ-; -biosis.
⋮
2. ···
⋮
408 Death ···
⋮
⋮
⋮
⋮

**Figure 1**
The structure of *Roget's Thesaurus*

| Lid | | |
|---|---|---|
| | clothing | 231.35 |
| | cover | 228.5 |
| | eyelid | 439.9 |
| | stopper | 266.4 |

**Figure 2**
Index entry for the word *lid*

verbs, adjectives, and so on.

The thesaurus has an index, which allows for retrieval of words related to a given one. For each entry, a list of words suggesting its various distinct subsenses is given, and a category or paragraph number for each of these. Figure 2 shows the index entry for *lid*. To find words related to *lid* in its sense of *cover*, one would turn to paragraph 5 of category 228. An index entry may be a pointer to a category or paragraph if there are no subsenses to be distinguished.

### 2.2 Differences from Traditional Knowledge Bases

In the structure of traditional artificial intelligence knowledge bases, such as frames or semantic networks, words or ideas that are related are actually "physically close" in the representation. In a thesaurus this need not be true. Physical closeness has some importance, as can be seen clearly from the hierarchy, but words in the index of the thesaurus often have widely scattered categories, and each category often points to a widely scattered selection of categories.

The thesaurus simply groups words by idea. It does not have to name or classify the idea or relationship. In traditional knowledge bases, the relationships must be named. For example, in a semantic net, a relationship might be **isa** or **color-of**, and in a frame database, there might be a slot for **color** or **location**.

In Section 1, different types of word relationships were discussed: systematic semantic, nonsystematic semantic, word association, and words related by a common situation. A factor common to all but situational relationships is that there is a strong tendency for the word relationships to be captured in the thesaurus. This holds even for the nonsystematic semantic relations, which are the most problematic by definition. A thesaurus simply groups related words without attempting to explicitly name each relationship. In a traditional computer database, a systematic semantic relationship can be represented by a slot value for a frame, or by a named link in a semantic network. If it is hard to classify a relationship in a systematic semantic way, it will be hard to represent the relationship in a traditional frame or semantic network formalism. Of the 16 nonsystematic semantic lexical chains given as examples in Halliday and Hasan (1976), 14 were found in *Roget's Thesaurus* (1977) using the relations given in Section 3.2.2. This represents an 87% hit rate (but not a big sample space). Word associations show a strong tendency to be findable in a thesaurus. Of the 16 word association pairs given in Hirst (1987), 14 were found in *Roget's Thesaurus* (1977). Since two of the word senses were not contained in the thesaurus at all, this represents a 100% hit rate among those that were. Situational word relationships are not as likely to be found in a general thesaurus. An example of a situational relationship is between *car* and *lights*, where the two words are clearly related in the situation involving a car's lights, but the relationship will not be found between them in a general thesaurus.

## 3. Finding Lexical Chains

### 3.1 General Methodology

We now describe a method of building lexical chains for use as an aid in determining the structure of text. This section details how these lexical chains are formed, using a thesaurus as the main knowledge base. The method is intended to be useful for text in any general domain. Unlike methods that depend on a full understanding of text, our method is the basis of a computationally feasible approach to determining discourse structure.

We developed our method in the following way. First, we took five texts, totaling 183 sentences, from general-interest magazines (*Reader's Digest, Equinox, The New Yorker, Toronto,* and *The Toronto Star*). Using our intuition (i.e., common sense and a knowledge of English), we identified the lexical chains in each text. We then formalized our intuitions into an algorithm, using our experience with the texts to set values for the following parameters (to be discussed below).

- thesaural relations
- transitivity of word relations
- distance (in sentences) allowable between words in a chain

The aim was to find efficient, plausible methods that will cover enough cases to ensure the production of meaningful results.

### 3.2 Forming Lexical Chains

**3.2.1 Candidate Words.** The first decision in lexical chain formation is which words in the text are candidates for inclusion in chains. As pointed out by Halliday and Hasan (1976), repetitive occurrences of closed-class words such as pronouns, prepositions, and verbal auxiliaries are obviously not considered. Also, high-frequency words like *good, do,* and *taking* do not normally enter into lexical chains (with some exceptions such as *takings* used in the sense of *earnings*). For example, in (13) only the italicized words should be considered as lexical chain candidates:

**Example 13**

My *maternal grandfather lived* to be *111. Zayde* was *lucid* to the *end,* but a few *years before* he *died* the *family assigned* me the *task* of *talking* to him about his *problem* with *alcohol.*

It should be noted that morphological analysis on candidate words was done intuitively, and would actually have to be formally implemented in an automated system.

**3.2.2 Building Chains.** Once the candidate words are chosen, the lexical chains can be formed. For this work an abridged version of *Roget's Thesaurus* (1977) was used. The chains were built by hand. Automation was not possible, for lack of a machine-readable copy of the thesaurus. Given a copy, implementation would clearly be straightforward. It is expected that research with an automated system and a large sample space of text would give valuable information on the fine-tuning of the parameter settings used in the general algorithm.

Five types of thesaural relations between words were found to be necessary in forming chains, but two (the first two below) are by far the most prevalent, constituting

over 90% of the lexical relationships. The relationships are the following:

1. Two words have a category common in their index entries. For example, *residentialness* and *apartment* both have category 189 in their index entries (see Figure 3.1).
2. One word has a category in its index entry that contains a pointer to a category of the other word. For example *car* has category 273 in its index entry, and that contains a pointer to category 276, which is a category of the word *driving* (see Figure 3.2).
3. A word is either a label in the other word's index entry (see Figure 3.3b), or is in a category of the other word. For example, *blind* has category 442 in its index entry, which contains the word *see* (see Figure 3.3a).
4. Two words are in the same group, and hence are semantically related. For example, *blind* has category 442, **blindness**, in its index entry and *see* has category 441, **vision**, in its index entry (see Figure 3.4).
5. The two words have categories in their index entries that both point to a common category. For example, *brutal* has category 851, which in turn

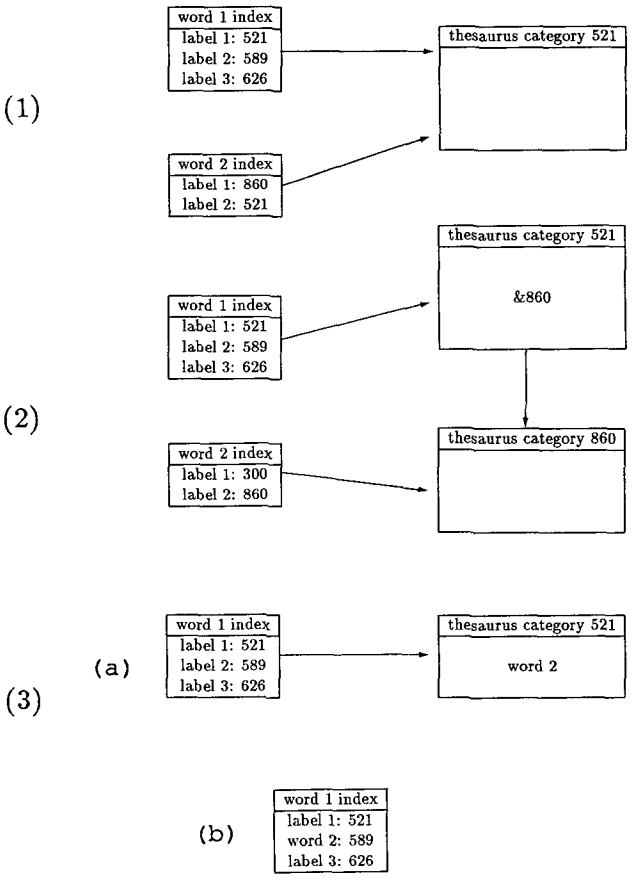
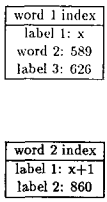



**Figure 3**
Thesaural Relations, parts (1)–(3)

(4)




(5)

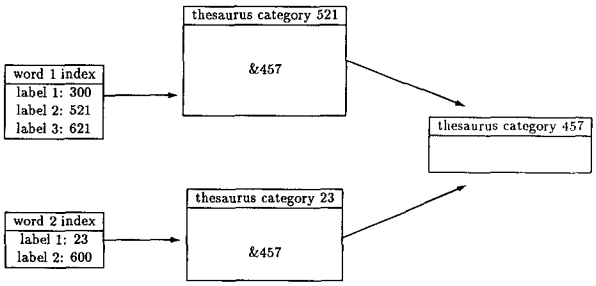



**Figure 3**
**Continued.** Thesaural Relations, parts (4)–(5)

has a pointer to category 830. *Terrified* has category 860 that likewise has a pointer to category 830 (see Figure 3.5).

One must consider how much transitivity to use when computing lexical chains. Specifically, if word *a* is related to word *b*, word *b* is related to word *c*, and word *c* is related to word *d* then is word *a* related to words *c* and *d*?

Consider this chain: {*cow, sheep, wool, scarf, boots, hat, snow*}. If unlimited transitivity were allowed, then *cow* and *snow* would be considered related, which is definitely counter intuitive. Our intuition was to allow one transitive link: word *a* is related to word *c* but not to word *d*. It seemed that two or more transitive links would so severely weaken the word relationship as to cause it to be nonintuitive. Our analysis of our sample texts supported this. To summarize, a transitivity of one link is sufficient to successfully compute the intuitive chains. An automated system could be used to test this out extensively, varying the number of transitive links and calculating the consequences. It is likely that it varies slightly with respect to style, author, or type of text.

There are two ways in which a transitive relation involving one link can cause two words to be related. In the first way, if word *a* is related to word *b*, and word *b* is related to word *c*, then word *a* is related to word *c*. In the second way, if word *a* is related to word *b*, and word *a* is related to word *c*, then word *b* is related to word *c*. But lexical chains are calculated only with respect to the text read so far. For example, if word *c* is related to word *a* and to word *b*, then word *a* and word *b* are not related, since at the time of processing, they were not relatable. Symmetry was not found to be necessary for computing the lexical chains.

We now consider how many sentences can separate two words in a lexical chain before the words should be considered unrelated. Now, sometimes, several sentences after a chain has clearly stopped, it is returned to. Such *chain returns* link together larger expanses of text than are contained in single chains or *chain segments*. Returns

to existing chains often correspond to intentional boundaries, as they occur after digressions or subintentions, thereby signalling a resumption of some structural text entity.

Intuitively, the distance between words in a chain is a factor in chain formation. The distance will not be "large," because words in a chain co-relate due to recognizable relations, and large distances would interfere with the recognition of relations.

The five texts were analyzed with respect to distance between clearly related words. The analysis showed that there can be up to two or three intermediary sentences between a word and the preceding element of a chain segment with which it can be linked. At distances of four or more intermediary sentences, the word is only able to signal a return to an existing chain. Returns happened after between 4 and 19 intermediary sentences in the sample texts. One significant fact emerged from this analysis: returns consisting of one word only were always made with a repetition of one of the words in the returned-to chain. Returns consisting of more than one word did not necessarily use repetition — in fact in most cases, the first word in the return was not a repetition.

The question of chain returns and when they can occur requires further research. When distances between relatable words are not tightly bound (as in the case of returns), the chances of incorrect chain linkages increase. It is anticipated that chain return analysis would become integrated with other text processing tools in order to prevent this. Also, we believe that chain *strength* analysis will be required for this purpose. Intuitively, some lexical chains are "stronger" than others, and possibly only strong chains can be returned to. There are three factors contributing to chain strength.

1. Reiteration — the more repetitions, the stronger the chain.
2. Density — the denser the chain, the stronger it is.
3. Length — the longer the chain, the stronger it is.

Ideally, some combination of values reflecting these three factors should result in a chain strength value that can be useful in determining whether a chain is strong enough to be returned to. Also, a strong chain should be more likely to have a structural correspondence than a weak one. It seems likely that chains could contain particularly strong portions with special implications for structure. These issues will not be addressed here.

**3.2.3 Notation and Data Structures.** In the computation of lexical chains, the following information is kept for each word in a chain:

- A word number, which is a sequential, chain-based number for each word so that it can be uniquely identified.
- The sentence number in which the word occurs.
- The chain created so far.

Each lexical relationship in a chain is represented as $(u,v)_x^y$ where:

- $u$ is the current word number,
- $v$ is the word number of the related word,
- $x$ is the transitive distance:

| Chain 1 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. evade | 15 | |
| 2. feigning | 15 | $(2,1)_0^2$ |
| 3. escaped | 16 | $(3,1)_1^0\ (3,2)_1^{T1}$ |

**Figure 4**
Lexical chain notation

- — **0** means no transitive link was used to form the word relationship
- — **1** means one transitive link was used to form the word relationship

- $y$ is either
  - — the number of the thesaural relationship between the two words (as given in Section 3.2.2)
  - — $Tq$ where
    - * $T$ stands for transitively related
    - * $q$ is the word number through which the transitive relation is formed.

A full example of this notation is shown in Figure 4.

Figure 5 shows the generalized algorithm for computing lexical chains. The parameter values that we used are shown for the following:

- candidate words
- thesaural relations
- transitivity of word relations
- distance between words in a chain.

The only parameter not addressed in this work is which (if any) chains should be eliminated from the chain-finding process.

### 3.3 Problems and Concerns

This section is a discussion of problems encountered during the computation of the lexical chains contained in our corpus of texts. The text example used in this paper is in Section 4.2, and the chains found in the example are in Appendix A.

**3.3.1 Where the Thesaurus Failed to Find Lexical Relations.** The algorithm found well over 90% of the intuitive lexical relations in the five examples we studied. The following is an analysis of when the thesaurus failed to find a relationship and why.

One problem was when the relationship between words was due more to their "feel" than their meaning. For example, in chain 6, the intuitive chain {*hand-in-hand*, matching, whispering, *laughing*, *warm*} was not entirely computable. Only the italicized words were relatable. The words in chain 6 are cohesive by virtue of being general, but strong, "good" words related by their goodness, rather than by their specific meanings. Chain 10, {*environment*, *setting*, *surrounding*}, was not thesaurally relatable. *Setting* was

```
REPEAT
        READ next word
        IF word is suitable for lexical analysis (see section 3.2.1) THEN
                CHECK for chains within a suitable span
                (up to 3 intermediary sentences, and no limitation on
                returns):
                        CHECK thesaurus for relationships (section 3.2.2).
                        CHECK other knowledge sources
                        (situational, general words, proper names).
                IF chain relationship is found THEN
                        INCLUDE word in chain.
                        CALCULATE chain so far
                        (allow one transitive link).
                END IF
                IF there are words that have not formed a chain for a suitable
                number of sentences (up to 3) THEN
                        ELIMINATE words from the span.
                END IF
                CHECK new word for relevance to existing chains that
                are suitable for checking.
                ELIMINATE chains that are not suitable for checking.
        END IF
END REPEAT
```

**Figure 5**
Algorithm for Finding Lexical Chains

not in the thesaurus, and while it seems as though *environment* and *surrounding* should be thesaurally connected, they were not.

Place names, street names, and people's names are generally not to be found in *Roget's Thesaurus* (1977). However, they are certainly contained in one's "mental thesaurus." Chain 1, which contains several major Toronto street names, is a good example of this. These names were certainly related to the rest of chain 1 in the authors' mental thesaurus, since we are residents of Toronto (and indeed the article assumed a knowledge of the geography of the city). In chain 5, the thesaurus did not connect the words *pine* and *trunk* with the rest of the chain {*virgin, bush, trees, trees*}. In a general thesaurus, specific information on, and classification of, plants, animals, minerals, etc., is not available.

To summarize, there were few cases in which the thesaurus failed to confirm an intuitive lexical chain. For those cases in which the thesaurus did fail, three missing knowledge sources became apparent.

1. General semantic relations between words of similar "feeling."
2. Situational knowledge.
3. Specific proper names.

**3.3.2 Problems with Distances and Chain Returns.** Occasionally the algorithm would cause two chains to merge together, whereas intuition would lead one to keep them

separate. We found the following intuitively separate chain beginning in sentence 38: {*people, Metropolitan Toronto, people, urban, population, people, population, population, people*}. However, the algorithm linked this chain with chain 1, which runs through the entire example and consists of these words and others: {*city, suburbs, traffic, community*}. Fortunately, this was a rare occurrence. But note that there will be cases in which lexical chains should be merged as a result of the intentional merging of ideas or concepts in the text.

Conversely, there were a few cases of unfortunate chain returns occurring where they were definitely counter intuitive. In chain 3, word 4, *wife*, was taken as a one-word return to the chain {*married, wife, wife*}. However, there is no intuitive reason for this.

## 4. Using Lexical Chains to Determine Text Structure

This section describes how lexical chains formed by the algorithm given in Section 3.2.3 can be used as a tool.

### 4.1 Lexical Chains and Text Structure

Any structural theory of text must be concerned with identifying units of text that are about the same thing. When a unit of text is about the same thing there is a strong tendency for semantically related words to be used within that unit. By definition, lexical chains are chains of semantically related words. Therefore it makes sense to use them as clues to the structure of the text.

This section will concentrate on analyzing correspondences between lexical chains and structural units of text, including:

- the correspondence of chain boundaries to structural unit boundaries;
- returns to existing chains and what they indicate about structural units;
- lexical chain strength and reliability of predicting correspondences between chains and structural units;
- an analysis of problems encountered, and when extra textual information is required to validate the correspondences between lexical chains and structural components.

The text structure theory chosen for this analysis was that of Grosz and Sidner (1986). It was chosen because it is an attempt at a general domain-independent theory of text structure that has gained a significant acceptance in the field as a good standard approach.

The methodology we used in our analyses was as follows:

1. We determined the lexical chain structure of the text using the algorithm given in Section 3.2.3. (In certain rare cases where the algorithm did not form intuitive lexical chains properly, it is noted, both in Section 3.4 and in the analysis in this section. The intuitive chain was used for the analysis; however the lexical chain data given in Appendix A show the rare mismatches between intuition and the algorithm.)
2. We determined the *intentional structure* of the text using the theory outlined by Grosz and Sidner.

3. We compared the lexical structure formed in step 1 with the intentional structure formed in step 2, and looked for correspondences between them.

### 4.2 An Example

Example 14 shows one of the five texts that we analyzed. It is the first section of an article in *Toronto* magazine, December 1987, by Jay Teitel, entitled "Outland."[2] The tables in Appendix A show the lexical chains for the text. (The other four texts and their analyses are given in Morris 1988.)

**Example 14**

1. ¶I spent the first 19 years of my life in the suburbs, the initial 14 or so relatively contented, the last four or five wanting mainly to be elsewhere.
2. The final two I remember vividly: I passed them driving to and from the University of Toronto in a red 1962 Volkswagen 1500 afflicted with night blindness.
3. The car's lights never worked — every dusk turned into a kind of medieval race against darkness, a panicky, mournful rush north, away from everything I knew was exciting, toward everything I knew was deadly.
4. I remember looking through the windows at the commuters mired in traffic beside me and actively hating them for their passivity.
5. I actually punched holes in the white vinyl ceiling of the Volks and then, by way of penance, wrote beside them the names and phone numbers of the girls I would call when I had my own apartment in the city.
6. One thing I swore to myself: I would never live in the suburbs again.
7. ¶My aversion was as much a matter of environment as it was traffic — one particular piece of the suburban setting: the "cruel sun."
8. Growing up in the suburbs you can get used to a surprising number of things — the relentless "residentialness" of your surroundings, the weird certainty you have that everything will stay vaguely new-looking and immune to historic soul no matter how many years pass.
9. You don't notice the eerie silence that descends each weekday when every sound is drained out of your neighbourhood along with all the people who've gone to work.
10. I got used to pizza, and cars, and the fact that the cultural hub of my community was the collective TV set.
11. But once a week I would step outside as dusk was about to fall and be absolutely bowled over by the setting sun, slanting huge and cold across the untreed front lawns, reminding me not just how barren and sterile, but how undefended life could be.
12. As much as I hated the suburban drive to school, I wanted to get away from the cruel suburban sun.
13. ¶When I was married a few years later, my attitude hadn't changed.
14. My wife was a city girl herself, and although her reaction to the suburbs was less intense than mine, we lived in a series of apartments safely straddling Bloor Street.
15. But four years ago, we had a second child, and simultaneously the school my wife taught at moved to Bathurst Street north of Finch Avenue.

2 © Jay Teitel. Reprinted with kind permission of the author.

16. She was now driving 45 minutes north to work every morning, along a route that was perversely identical to the one I'd driven in college.
17. ¶We started looking for a house.
18. Our first limit was St. Clair — we would go no farther north.
19. When we took a closer look at the price tags in the area though, we conceded that maybe we'd have to go to Eglinton — but that was definitely it.
20. But the streets whose names had once been magical barriers, latitudes of tolerance, quickly changed to something else as the Sundays passed.
21. Eglinton became Lawrence, which became Wilson, which became Sheppard.
22. One wind-swept day in May I found myself sitting in a town-house development north of Steeles Avenue called Shakespeare Estates.
23. It wasn't until we stepped outside, and the sun, blazing unopposed over a country club, smacked me in the eyes, that I came to.
24. It was the cruel sun.
25. We got into the car and drove back to the Danforth and porches as fast as we could, grateful to have been reprieved.
26. ¶And then one Sunday in June I drove north alone.
27. This time I drove up Bathurst past my wife's new school, hit Steeles, and kept going, beyond Centre Street and past Highway 7 as well.
28. I passed farms, a man selling lobsters out of his trunk on the shoulder of the road, a chronic care hospital, a country club and what looked like a mosque.
29. I reached a light and turned right.
30. I saw a sign that said Houses and turned right again.
31. ¶In front of me lay a virgin crescent cut out of pine bush.
32. A dozen houses were going up, in various stages of construction, surrounded by hummocks of dry earth and stands of precariously tall trees nude halfway up their trunks.
33. They were the kind of trees you might see in the mountains.
34. A couple was walking hand-in-hand up the dusty dirt roadway, wearing matching blue track suits.
35. On a "front lawn" beyond them, several little girls with hair exactly the same colour of blond as my daughter's were whispering and laughing together.
36. The air smelled of sawdust and sun.
37. ¶It was a suburb, but somehow different from any suburb I knew.
38. It felt warm.
39. ¶It was Casa Drive.
40. ¶In 1976 there were 2,124,291 people in Metropolitan Toronto, an area bordered by Steeles Avenue to the north, Etobicoke Creek on the west, and the Rouge River to the east.
41. In 1986, the same area contained 2,192,721 people, an increase of 3 percent, all but negligible on an urban scale.
42. In the same span of time the three outlying regions stretching across the top of Metro — Peel, Durham, and York — increased in population by 55 percent, from 814,000 to some 1,262,000.
43. Half a million people had poured into the crescent north of Toronto in the space of a decade, during which time the population of the City of Toronto actually declined as did the populations of the "old" suburbs with the exception of Etobicoke and Scarborough.
44. If the sprawling agglomeration of people known as Toronto has boomed in the past 10 years it has boomed outside the traditional city confines in a totally new city, a new suburbia containing one and a quarter million people.

### 4.3 The Correspondences between Lexical and Intentional Structures

In Figure 6 we show the intentional structure of the text of Section 4.2, and in Figure 7 we show the correspondences between the lexical chains and intentions of the example.

There is a clear correspondence between chain 1, {..., *driving*, *car's*, ...}, and intention 1 (changing attitudes to suburban life). The continuity of the subject matter is reflected by the continuous lexical chain. From sentence 40 to sentence 44, two words, *population* and *people* are used repetitively in the chain. *Population* is repeated three times, and *people* is repeated five times. If chain strength (indicated by the reiteration) were used to delineate "strong" portions of a chain, this strength information could also be used to indicate structural attributes of the text. Specifically, sentences 40 to 44 form intention 1.3 (why new suburbs exist), and hence a strong portion of the

1 (1–44)
Changing attitudes to suburban life.

1.1 (1–25)
Earlier aversion to suburban life.

1.1.1 (1–7)
Hatred of commuting.

1.1.2 (8–12)
The hated suburb environment.

1.1.3 (13–25)
How this old aversion to suburbs held, when a recent attempt was made to buy a new house in the suburbs.

1.1.3.1 (13–16)
How life changed, giving author reason to look for a new house.

1.1.3.2 (17–22)
Houses are too expensive in Metro Toronto, hence one must look in the suburbs to buy a house.

1.1.3.3 (23–25)
The old familiar aversion to suburbs came back.

1.2 (26–39)
A new suburb that seems livable in and nice.

1.2.1 (26–30)
The drive to the new suburb.

1.2.2 (31–33)
The forested area.

1.2.3 (34–39)
The pleasant environment.

1.3 (40–44)
Why the new suburbs exist.

**Figure 6**
The Intentional Structure of Example 14 (showing topics the writer intends to discuss)

| Chain | Chain Range | Intention | Intention Range |
|---|---|---|---|
| 1 | 1–44 | 1 | 1–44 |
| 2.1 | 2–12 | 1.1.1, 1.1.2 | 1–12 |
| 2.2 | 16 | end of 1.1.3.1 | 16 |
| 2.3 | 24 | end of 1.1.3.3 | 25 |
| 3 | 13–15 | 1.1.3.1 | 13–16 |
| 4 | 19–20 | 1.1.3.2 | 17–22 |
| 5 | 31–33 | 1.2.2 | 31–33 |
| 6 | 34–38 | 1.2.3 | 34–39 |
| 7,8 | 1–3 | 1.1.1 | 1–7 |
| 9 | 7–8 | 1.1.2 | 8–12 |

**Figure 7**
Correspondences between lexical and intentional structures

chain would correspond exactly to a structural unit. In addition, *drive* was repeated eight times between sentence 2 and sentence 26, corresponding to intention 1.1 (earlier aversion to suburban life). *Suburb* was repeated eleven times throughout the entire example, indicating the continuity in structure between sentences 1–44.

Chain 2.1, {*afflicted, darkness, ...*}, from sentence 2 to sentence 12, corresponds to intentions 1.1.1 (hatred of commuting) and 1.1.2 (hatred of suburbs). More textual information is needed to separate intentions 1.1.1 and 1.1.2. There is a one-word return to chain 2 at sentences 16 and 24, strongly indicating that chain 2 corresponds to intention 1.1, which runs from sentence 1 to sentence 25. Also, segment 2.2 coincides with the end of intention 1.1.3.1 (how life changed), and segment 2.3 coincides with the end of intention 1.1.3.3 (old familiar aversion to suburbs). This situation illustrates how chain returns help indicate the structure of the text. If chain returns were not considered, chain 2 would end at sentence 12, and the structural implications of the two single-word returns would be lost. It is intuitive that the two words *perverse* and *cruel* indicate links back to the rest of intention 1.1. The link provided by the last return, *cruel*, is especially strong, since it occurs after the diversion describing the attempt to find a nice house in the suburbs. *Cruel* is the third reiteration of the word in chain 2.

Chain 3, {*married, wife, ...*}, corresponds to intention 1.1.3.1 (if the unfortunate chain return mentioned in section 3.4.2 is ignored) and chain 4 {*conceded, tolerance*}, corresponds to intention 1.1.3.2 (expensive houses in Metro Toronto). The boundaries of chain 4 are two sentences inside the boundaries of the intention. The existence of a lexical chain is a clue to the existence of a separate intention, and boundaries within one or two sentences of the intention boundaries are considered to be close matches.

Chain 5, {*virgin, pine, ...*}, corresponds closely to intention 1.2.2 (forested area). Chain 6, {*hand-in-hand, matching, ...*}, corresponds closely to intention 1.2.3 (pleasant environment). Chains 7, {*first, initial, final*}, and 8, {*night, dusk, darkness*}, are a couple of short chains (three words long) that overlap. They collectively correspond to intention 1.1.1 (hatred of commuting). The fact that they are short and overlapping suggests that they could be taken together as a whole.

Chain 9, {*environment, setting, surrounding*}, corresponds to intention 1.1.2 (hated suburbs). Even though the chain is a lot shorter in length than the intention, its presence is a clue to the existence of a separate intention in its textual vicinity. Since the lexical chain boundary is more than two sentences away from the intention boundary, other textual information would be required to confirm the structure.

Overall, the lexical chains found in this example provide a good clue for the determination of the intentional structure. In some cases, the chains correspond exactly to an intention. It should also be stressed, however, that the lexical structures cannot be used on their own to predict an exact structural partitioning of the text. This of course was never expected. As a good example of the limitations of the tool, intention 1.2 (nice new suburb) starts in sentence 26, but there are no new lexical chains starting there. The only clue to the start of the new intention would be the ending of chain 2 {*afflicted, darkness, ...*}.

This example also provides a good illustration (chain 2) of the importance of chain returns being used to indicate a high-level intention spanning the length of the entire chain (including all segments). Also, the returns coincided with intentional boundaries.

## 5. Conclusions

The motivation behind this work was that lexical cohesion in text should correspond in some way to the structure of the text. Since lexical cohesion is a result of a unit of text being, in some recognizable semantic way, about a single topic, and text structure

analysis involves finding the units of text that are about the same topic, one should have something to say about the other. This was found to be true. The lexical chains computed by the algorithm given in Section 3.2.3 correspond closely to the intentional structure produced from the structural analysis method of Grosz and Sidner (1986). This is important, since Grosz and Sidner give no method for computing the intentions or linguistic segments that make up the structure that they propose.

Hence the concept of lexical cohesion, defined originally by Halliday and Hasan (1976) and expanded in this work, has a definite use in an automated text understanding system. Lexical chains are shown to be almost entirely computable with the relations defined in Section 3.2.2. The computer implementation of this type of thesaurus access would be a straightforward task involving traditional database techniques. The program to implement the algorithm given in Section 3.2.3 would also be straightforward. However, automated testing could help fine-tune the parameters, and would help to indicate any unfortunate chain linkages. Although straightforward from an engineering point of view, the automation would require a significant effort. A machine-readable thesaurus with automated index searching and lookup is required.

The texts we have analyzed, here and elsewhere (Morris 1988) are general-interest articles taken from magazines. They were chosen specifically to illustrate that lexical cohesion, and hence this tool, is not domain-specific.

### 5.1 Improvements on Earlier Research

The methods used in this work improve on those from Halliday and Hasan (1976). Halliday and Hasan related words back to the first word to which they are tied, rather than forming explicit lexical chains that include the relationships to intermediate words in the chain. They had no notions of transitivity, distance between words in a chain, or chain returns. Their intent was not a computational means of finding lexical chains, and they did not suggest a thesaurus for this purpose.

Ventola (1987) analyzed lexical cohesion and text structure within the framework of systemic linguistics and the specific domain of service encounters such as the exchange of words between a client at a post office and a postal worker. Ventola's chain-building rule was that each lexical item is "taken back once to the nearest preceding lexically cohesive item regardless of distance" (p. 131). In our work the related words in a chain are seen as indicating structural units of text, and hence distance between words is relevant. Ventola did not have the concept of chain returns, and transitivity was allowed up to any level. Her research was specific to the domain used. She does not discuss a computational method of determining the lexical chains.

Hahn (1985) developed a text parsing system that considers lexical cohesion. Nouns in the text are mapped directly to the underlying model of the domain, which was implemented as a frame-structured knowledge base. Hahn viewed lexical cohesion as a local phenomenon between words in a sentence and the preceding one. There was also an extended recognizer that worked for cohesion contained within paragraph boundaries. Recognizing lexical cohesion was a matter of searching for ways of relating frames and slots in the database that are activated by words in the text. Heavy reliance is put on the "formally clear cut model of the underlying domain" (Hahn 1985, p. 3). However, general-interest articles such as we analyzed do not have domains that can be *a priori* formally represented as frames with slot values in such a manner that lexical cohesion will correspond directly to them. Our work uses lexical cohesion as it naturally occurs in domain-independent text as an indicator of unity, rather than fitting a domain model to the lexical cohesion. Hahn does not use the concept of chain returns or transitivity.

Sedelow and Sedelow (1986, 1987) have done a significant amount of research

on the thesaurus as a knowledge source for use in a natural language understanding system. They have been interested in the application of clustering patterns in the thesaurus. Their student Bryan (1973) proposed a graph-theoretic model of the thesaurus. A boolean matrix is created with words on one axis and categories on the other. A cell is marked as true if a word associated with a cell intersects with the category associated with a cell. Paths or chains in this model are formed by traveling along rows or columns to other true cells. Semantic "neighborhoods" are grown, consisting of the set of chains emanating from an entry. It was found that without some concept of chain strength, the semantic relatedness of these neighborhoods decays, partly due to homographs. Strong links are defined in terms of the degree of overlap between categories and words. A strong link exists where at least two categories contain more than one word in common, or at least two words contain more than one category in common. The use of strong links was found to enable the growth of strong semantic chains with homograph disambiguation.

This concept is different from that used in our work. Here, by virtue of words co-occurring in a text and then also containing at least one category in common or being in the same category, they are considered lexically related and no further strength is needed. We use the thesaurus as a validator of lexical relations that are possible due to the semantic relations among words in a text.

### 5.2 Further Research

It has already been mentioned that the concept of chain strength needs much further work. The intuition is that the stronger a chain, the more likely it is to have a corresponding structural component.

The integration of this tool with other text understanding tools is an area that will require a lot of work. Lexical chains do not always correspond exactly to intentional structure, and when they do not, other textual information is needed to obtain the correct correspondences. In the example given, there were cases where a lexical chain did correspond to an intention, but the sentences spanned by the lexical chain and the intention differed by more than two. In these cases, verification of the possible correspondence must be accomplished through the use of other textual information such as semantics or pragmatics. Cue words would be interesting to address, since such information seems to be more computationally accessible than underlying intentions.

It would be useful to automate this tool and run a large corpus of text through it. We suspect that the chain-forming parameter settings (regarding transitivity and distances between words) will be shown to vary slightly according to author's style and the type of text. As it is impossible to do a complete and error-free lexical analysis of large text examples in a limited time-frame, automation is desirable. It could help shed some light on possible unfortunate chain linkages. Do they become problematic, and if so, when does this tend to happen? Research into limiting unfortunate linkages and detecting when the method is likely to produce incorrect results should be done (cf. Charniak 1986).

Analysis using different theories of text structure was not done, but could prove insightful. The independence of different people's intuitive chains and structure assignments was also not addressed by this paper.

A practical limitation of this work is that it depends on a thesaurus as its knowledge base. A thesaurus is as good as the work that went into creating it, and also depends on the perceptions, experience, and knowledge of its creators. Since language is not static, a thesaurus would have to be continually updated to remain current. Furthermore, no one thesaurus exists that meets all needs. *Roget's Thesaurus*, for example, is a general thesaurus that does not contain lexical relations specific to the geography

of Africa or quantum mechanics. Therefore, further work needs to be done on identifying other sources of word knowledge, such as domain-specific thesauri, dictionaries, and statistical word usage information, that should be integrated with this work. As an anonymous referee pointed out to us, *Volks* and *Volkswagen* were not included in the chain containing *driving* and *car*. These words were not in a general thesaurus, and were also missed by the authors!

Section 1 mentioned that lexical chains would be also useful in providing a context for word sense disambiguation and in narrowing to specific word meanings. As an example of a chain providing useful information for word sense disambiguation, consider words 1 to 15 of chain 2.1 of the example: {*afflicted, darkness, panicky, mournful, exciting, deadly, hating, aversion, cruel, relentless, weird, eerie, cold, barren, sterile, ...*}. In the context of all of these words, it is clear that *barren* and *sterile* do not refer to an inability to reproduce, but to a *cruel cold*ness. The use of lexical chains for ambiguity resolution is a promising area for further research.

**Acknowledgments**

Thanks to Robin Cohen, Jerry Hobbs, Eduard Hovy, Ian Lancashire, and anonymous referees for valuable discussions of the ideas in this paper. Thanks to Chrysanne DiMarco, Mark Ryan, and John Morris for commenting on earlier drafts. This work was financially assisted by the Government of Ontario, the Department of Computer Science of the University of Toronto, and the Natural Sciences and Engineering Research Council of Canada. We are grateful to Jay Teitel for allowing us to reprint text from his article "Outland."

**References**

Bryan, Robert M. (1973). "Abstract thesauri and graph theory applications to thesaurus research," in *Automated language analysis*, edited by Sally Yeates Sedelow, University of Kansas.

Carroll, Lewis (1872). *Through the Looking Glass*.

Charniak, Eugene (1986). "A neat theory of marker parsing." In *Proceedings, 5th National Conference on Artificial Intelligence*, Philadelphia, August 1986, 584–588.

Grosz, Barbara and Sidner, Candance (1986). "Attention, intentions and the structure of discourse." *Computational Linguistics*, 12(3), 175–204.

Hahn, Udo (1985). "On lexically distributed text parsing. A computational model for the analysis of textuality on the level of text cohesion and text coherence." In *Linking in text*, edited by Ferenc Kiefer, Universität Konstanz.

Halliday, Michael and Hasan, Ruqaiya (1976). *Cohesion in English*. Longman Group.

Hirst, Graeme (1987). *Semantic Interpretation and the Resolution of Ambiguity*. Studies in Natural Language Processing. Cambridge University Press.

Hirst, G. (1981). *Anaphora in Natural Language Understanding: A Survey*. Lecture Notes in Computer Science. Springer Verlag.

Hobbs, Jerry (1978). "Coherence and coreference." Technical note 168, SRI International.

McKeown, K. (1985). *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Studies in Natural Language Processing. Cambridge University Press.

Morris, Jane (1988). "Lexical cohesion, the thesaurus, and the structure of text." Technical report CSRI-219, Department of Computer Science, University of Toronto.

Postman, Leo and Keppel, Geoffrey, editors (1970). *Norms of Word Association*. Academic Press.

Reichman, Rachel (1985). *Getting Computers to Talk Like You and Me: Discourse Context, Focus, and Semantics (An ATN Model)*. The MIT Press.

Roget, P. (1977). *Roget's International Thesaurus, Fourth Edition*. Harper and Row Publishers Inc.

Sedelow, Sally and Sedelow, Walter (1987). "Semantic space." *Computers and translation*, 2, 235–245.

Sedelow, Sally and Sedelow, Walter (1986). "Thesaural knowledge representation." In *Proceedings, 2nd Annual Conference of the University of Waterloo Centre for the New Oxford English Dictionary: Advances in Lexicology*. University of Waterloo.

Ventola, E. (1987). *The Structure of Social Interaction: A Systemic Approach to the Semiotics of Service Encounters*. Open Linguistic Series. Frances Pinter Publishers.

## Appendix A

| Chain 1 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. suburbs | 1 | |
| 2. driving | 2 | |
| 3. Volkswagen | 2 | |
| 4. car's | 3 | $(4, 2)^{2}_{0}$ |
| 5. lights | 3 | |
| 6. commuters | 4 | |
| 7. traffic | 4 | $(7, 2)^{2}_{0}$ $(7, 4)^{1}_{0}$ |
| 8. Volks | 5 | |
| 9. apartment | 5 | $(9, 1)^{1}_{0}$ |
| 10. city | 5 | $(10, 1)^{1}_{0}$ $(10, 2)^{1}_{0}$ $(10, 4)^{T2}_{0}$ $(10, 7)^{1}_{0}$ $(10, 9)^{1}_{0}$ |
| 11. suburbs | 6 | $(11, 1)^{0}_{0}$ $(11, 9\text{-}10)^{1}_{0}$ $(11, 2\text{-}7)^{T10}_{1}$ |
| 12. traffic | 7 | $(12, 2)^{2}_{0}$ $(12, 4\text{-}10)^{1}_{0}$ $(12, 7)^{0}_{0}$ $(12, 11)^{T10}_{2}$ |
| 13. suburban | 7 | $(13, 1\text{-}11)^{0}_{0}$ $(13, 9\text{-}10)^{1}_{0}$ $(13, 2\text{-}12)^{T10}_{1}$ |
| 14. suburbs | 8 | $(14, 1\text{-}11\text{-}13)^{0}_{0}$ $(14, 9\text{-}10\text{-}13)^{1}_{0}$ $(14, 2\text{-}12)^{T10}_{1}$ |
| 15. residentialness | 8 | $(15, 1\text{-}9\text{-}10\text{-}13\text{-}14)^{1}_{0}$ $(15, 2\text{-}7\text{-}12)^{T10}_{1}$ |
| 16. neighbourhood | 9 | $(16, 1\text{-}11\text{-}13\text{-}14)^{1}_{0}$ $(16, 9\text{-}10\text{-}13)^{T14}_{1}$ |
| 17. community | 10 | |
| 18. suburban | 12 | $(18, 1\text{-}11\text{-}13\text{-}14)^{0}_{0}$ $(18, 9\text{-}10\text{-}16)^{1}_{0}$ $(18, 2\text{-}12)^{T10}_{1}$ |
| 19. drive | 12 | $(19, 2)^{0}_{0}$ $(19, 7\text{-}10\text{-}12)^{1}_{0}$ $(19, 4)^{2}_{0}$ $(19, 1\text{-}9\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18)^{T10}_{1}$ $(20, 9\text{-}10\text{-}16)^{1}_{0}$ $(20, 2\text{-}12\text{-}19)^{T10}_{2}$ |
| 20. suburban | 12 | $(20, 1\text{-}11\text{-}13\text{-}14\text{-}18)^{0}_{0}$ $(20, 9\text{-}10\text{-}16)^{1}_{0}$ $(20, 2\text{-}12\text{-}19)^{T1}_{1}$ |
| 21. city | 14 | $(21, 10)^{0}_{0}$ $(21, 1\text{-}2\text{-}7\text{-}9\text{-}13\text{-}14\text{-}15\text{-}16\text{-}19)^{T10}_{1}$ $(21, 4\text{-}12)^{T19}_{1}$ |
| 22. suburbs | 14 | $(22, 1\text{-}11\text{-}13\text{-}14\text{-}18\text{-}20)^{0}_{0}$ $(22, 9\text{-}10\text{-}16\text{-}21)^{1}_{0}$ $(22, 2\text{-}12\text{-}19)^{T10}_{1}$ |
| 23. apartments | 14 | $(23, 9)^{0}_{0}$ $(23, 1\text{-}10\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18\text{-}20\text{-}21\text{-}22)^{1}_{0}$ $(23, 2\text{-}4\text{-}7\text{-}12\text{-}19)^{T21}_{1}$ |
| 24. Bloor St. | 14 | |
| 25. Bathurst St. | 15 | |
| 26. Finch St. | 15 | |
| 27. driving | 16 | $(27, 2\text{-}19)^{0}_{0}$ $(27, 7\text{-}10\text{-}12\text{-}21)^{1}_{0}$ $(27, 4)^{2}_{0}$ $(27, 1\text{-}9\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18\text{-}20\text{-}22\text{-}23)^{T10}_{2}$ |
| 28. route | 16 | $(28, 1\text{-}2\text{-}9\text{-}10\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18\text{-}19\text{-}20\text{-}21\text{-}22\text{-}23\text{-}27)^{2}_{0}$ $(28, 4\text{-}7\text{-}12)^{T27}_{1}$ |
| 29. driven | 16 | $(29, 2\text{-}19\text{-}27\text{-}29)^{0}_{0}$ $(29, 7\text{-}10\text{-}12\text{-}21)^{1}_{0}$ $(29, 4\text{-}28)^{2}_{0}$ $(29, 1\text{-}9\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18\text{-}20\text{-}22\text{-}23)^{T10}_{1}$ |
| 30. house | 17 | $(30, 1\text{-}9\text{-}10\text{-}11\text{-}13\text{-}14\text{-}15\text{-}16\text{-}18\text{-}20\text{-}21\text{-}22\text{-}23)^{1}_{0}$ $(30, 2\text{-}4\text{-}7\text{-}12\text{-}19\text{-}27\text{-}28\text{-}29)^{T10}_{1}$ |
| 31. St. Clair | 18 | |
| 32. Eglinton | 19 | |

| Chain 1 (continued) | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 33. streets | 20 | (33, 1-10-13-14-15-16-18-20-21-22-23-30)$^{1}_{0}$ (33, 2-4-7-12-19-27-28-29)$^{T10}_{1}$ |
| 34. Eglinton | 21 | |
| 35. Lawrence | 21 | |
| 36. Wilson | 21 | |
| 37. Sheppard | 21 | |
| 38. town-house | 22 | (38, 30)$^{0}_{0}$ (38, 1-10-13-14-15-16-18-20-21-22-23)$^{1}_{0}$ (38, 2-4-7-12-19-27-28-29-33)$^{T10}_{1}$ |
| 39. Steeles | 22 | |
| 40. car | 25 | (40, 2-19-27-29)$^{1}_{0}$ (40, 4-7-10-12-21-28)$^{T29}_{1}$ |
| 41. drove | 25 | (41, 2-19-27-29)$^{0}_{0}$ (41, 7-10-12-21)$^{1}_{0}$ (41, 4-28)$^{2}_{0}$ (41, 1-9-11-13-14-15-16-18-20-22-30-38)$^{T10}_{1}$ |
| 42. Danforth | 25 | |
| 43. porches | 25 | (43, 33)$^{1}_{0}$ (43, 1-4-10-13-14-15-18-20-21-22-23-30-38-40)$^{2}_{0}$ (43, 16)$^{T38}_{1}$ (43, 2-19-23-29)$^{T40}_{1}$ |
| 44. drove | 26 | (44, 2-19-27-29-41)$^{0}_{0}$ (44, 7-10-12-21)$^{1}_{0}$ (44, 4-28)$^{2}_{0}$ (44, 1-9-11-13-14-15-16-18-20-22-23-30-38)$^{T10}_{1}$ |
| 45. drove | 27 | (45, 2-19-27-29-41-44)$^{0}_{0}$ (45, 7-10-12-21)$^{1}_{0}$ (45, 4-28)$^{2}_{0}$ (45, 1-9-11-13-14-15-16-18-20-22-23-30-38)$^{T10}_{1}$ |
| 46. Bathurst | 27 | |
| 47. Steeles | 27 | |
| 48. Centre St. | 27 | |
| 49. Highway 7 | 27 | |
| 50. trunk | 28 | |
| 51. road | 28 | (51, 1-9-10-11-13-14-15-16-18-20-21-22-23-28-30-38)$^{1}_{0}$ (51, 43)$^{2}_{0}$ (51, 7)$^{T10}_{1}$ (51, 16)$^{T38}_{1}$ |
| 52. light | 29 | (52, 5)$^{0}_{0}$ |
| 53. turned | 29 | |
| 54. houses | 30 | (54, 30-38)$^{0}_{0}$ (54, 1-9-10-11-13-14-15-18-20-21-22-23-33-43-52)$^{1}_{0}$ (54, 16-28)$^{2}_{0}$ (54, 2-7-12-19-29-41-44)$^{T10}_{1}$ |
| 55. turned | 30 | (55, 53)$^{0}_{0}$ |
| 56. houses | 32 | (56, 30-38-54)$^{0}_{0}$ (56, 1-9-10-11-13-14-15-18-20-21-22-23-33-43-51)$^{1}_{0}$ (56, 16-28)$^{2}_{0}$ (56, 2-7-12-19-29-41-44)$^{T10}_{1}$ |
| 57. roadway | 34 | (57, 51)$^{0}_{0}$ (57, 1-9-10-11-13-14-15-16-18-20-21-22-23-28-30-38)$^{1}_{0}$ (57, 43)$^{2}_{0}$ (57, 7)$^{T10}_{1}$ (57, 16)$^{T38}_{1}$ |
| 58. lawn | 35 | (58, 1-9-10-11-13-14-15-18-20-21-22-23-30-33-38-43-51-54-56-57)$^{1}_{0}$ (58, 28)$^{5}_{0}$ (58, 2-12-19-27-29-41-44)$^{T10}_{1}$ (58, 16)$^{T56}_{1}$ |
| 59. suburb | 37 | (59, 1-11-13-14-18-20-22)$^{0}_{0}$ (59, 30-38-56)$^{1}_{0}$ (59, 9-10-15-21-23-33-43-51)$^{1}_{0}$ (59, 16-28)$^{2}_{0}$ (59, 2-7-12-19-29-41-44)$^{T10}_{1}$ |

| Chain 1 (continued) | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 60. suburb | 37 | (60, 1-11-13-14-18-20-22-59)$_0^0$ (60, 30-38-56)$_0^1$ (60, 9-10-15-21-23-33-43-51-54-56-57-59)$_0^1$ (60, 16-28)$_0^2$ (60, 2-7-12-19-29-41-4446-47)$_1^{T10}$ |
| 61. people | 40 | (61, 15)$_0^1$ (61, 1-9-10-11-13-14-18-20-21-22-23-30-33-38-51-54-56-57-59-60)$_0^2$ (61, 2-7-12-19-27-29-41-44)$_1^{T10}$ (61, 16-43-58)$_1^{T56}$ |
| 62. Metropolitan Toronto | 40 | (62, 1-9-10-11-13-14-15-18-20-21-22-23-30-33-38-51-54-56-57-59-60)$_0^1$ (62, 2-7-12-19-27-29-41-44)$_1^{T10}$ (62, 16-43-58)$_0^2$ |
| 63. Steeles | 40 | |
| 64. people | 41 | (64, 61)$_0^0$ (64, 15)$_0^1$ (64, 1-9-10-11-13-14-18-20-21-22-23-30-33-38-51-54-56-57-59-60-62)$_0^2$ (65, 2-7-12-19-27-29-41-44)$_1^{T10}$ (61, 16-43-58)$_1^{T56}$ |
| 65. urban | 41 | (65, 1-9-10-11-13-14-15-18-20-21-22-23-30-33-38-51-54-56-57-59-60-62)$_0^1$ (65, 2-7-12-19-27-29-41-44)$_1^{T10}$ (65, 16-43-58)$_0^2$ |
| 66. Metro | 42 | (66, 62)$_0^0$ (66, 1-9-10-11-13-14-15-18-20-21-22-23-30-33-38-51-54-56-57-59-60)$_0^1$ (66, 2-7-12-19-27-29-41-44)$_1^{T10}$ (66, 16-43-58-64)$_0^2$ |
| 67. Peel | 42 | |
| 68. Durham | 42 | |
| 69. York | 42 | |
| 70. population | 42 | (70, 30-38-54-56-61-64)$_0^1$ (70, 1-9-10-11-13-14-15-18-20-21-22-23-33-51-57-59-60-62-65-66)$_0^2$ (70, 43-58)$_0^5$ (70, 2-7-12-19-27-29-41-44)$_1^{T10}$ (70, 16)$_1^{T64}$ |
| 71. people | 43 | (71, 61-64)$_0^0$ (71, 15-70)$_0^1$ (71, 1-9-10-11-13-14-18-20-21-22-23-30-33-38-51-54-56-57-59-60-62-65-66)$_0^2$ (71, 2-7-12-19-27-29-41-44)$_1^{T10}$ (71, 16-43-58-64)$_1^{T56}$ |
| 72. Toronto | 43 | |
| 73. population | 43 | (73, 70)$_0^0$ (73, 30-38-51-54-56-61-65-71)$_0^1$ (73, 1-9-10-11-13-14-15-18-20-21-22-23-33-51-57-59-60-62-65-66)$_0^2$ (73, 43-58)$_0^5$ (73, 2-7-12-19-27-29-41-44)$_1^{T10}$ (73, 16)$_1^{T64}$ |
| 74. city | 43 | (74, 10-21)$_0^0$ (74, 1-2-7-9-11-12-13-14-15-18-19-20-22-23-27-29-30-33-38-41-44-51-54-56-57-59-60-62-65)$_0^1$ (74, 16-28-43-58-65-70-71-73)$_0^2$ (74, 4-40)$_1^{T47}$ |
| 75. Toronto | 43 | |
| 76. population | 43 | (76, 70-73)$_0^0$ (76, 30-38-54-56-61-64-71)$_0^1$ (76, 1-9-10-11-13-14-15-18-20-21-22-23-33-51-57-59-60-62-65-66-74)$_0^2$ (76, 43-58)$_0^5$ (76, 2-7-12-19-27-29-41-44)$_1^{T10}$ (76, 16)$_1^{T64}$ |
| 77. suburbs | 43 | (77, 1-11-13-14-18-20-22-59-60)$_0^0$ (77, 30-38-56-62-65-66-74)$_0^1$ (77, 9-10-15-21-23-33-43-51)$_0^1$ (77, 16-28-64-70-71-72-73-76)$_0^2$ (77, 2-7-12-19-29-41-44-)$_1^{T10}$ |
| 78. Etobicoke | 43 | |

| Chain 1 (continued) | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 79. Scarborough | 43 | |
| 80. people | 44 | (80, 61-64-71)$_0^0$ (80, 15-70)$_0^1$ (80, 1-9-10-11-13-14-18-20-21-22-23-30-33-38-51-54-56-57-59-60-62-65-66-73-76-77)$_0^2$ (80, 2-7-12-19-27-29-41-44)$_1^{T10}$ (80, 16-43-58)$_1^{T56}$ |
| 81. Toronto | 44 | |
| 82. city | 44 | (82, 10-21-74)$_0^0$ (82, 1-2-7-9-11-12-13-14-15-18-19-20-22-23-27-29-30-33-38-41-44-46-47-51-54-56-57-59-60-62-65-77)$_0^1$ (82, 16-28-43-58-64-70-71-73-76-80)$_0^2$ (82, 4-40)$_1^{T47}$ |
| 83. suburbia | 44 | (83, 1-11-13-14-18-20-22-59-60-77)$_0^0$ (83, 30-38-56-82)$_0^1$ (83, 9-10-15-21-23-33-43-51-82)$_0^1$ (83, 16-28-80)$_0^2$ (83, 2-7-12-19-29-41-44)$_1^{T10}$ |
| 84. people | 44 | (84, 61-64-71-80)$_0^0$ (84, 15-70-82)$_0^1$ (84, 1-9-10-11-13-14-18-20-21-22-23-30-33-38-51-54-56-57-59-60-62-65-66-73-76-77-82)$_0^2$ (84, 2-7-12-19-27-29-41-44)$_1^{T10}$ (84, 16-43-58)$_1^{T56}$ |

| Chain 2, Segment 1 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. afflicted | 2 | |
| 2. darkness | 3 | (2, 1)$_0^2$ |
| 3. panicky | 3 | (3, 1)$_0^2$ (3, 2)$_0^5$ |
| 4. mournful | 3 | (4, 1)$_0^1$ (4, 2)$_0^1$ (4, 3)$_0^2$ |
| 5. exciting | 3 | (5, 1-4)$_0^2$ (5, 2-3)$_0^5$ |
| 6. deadly | 3 | (6, 1-4)$_0^2$ (6, 2-3-5)$_0^5$ |
| 7. hating | 4 | (7, 1-4)$_0^1$ (7, 2-3-5-6)$_0^2$ |
| 8. aversion | 7 | (8, 7)$_0^1$ (8, 1-4)$_0^2$ (8, 2-3-5-6)$_0^5$ |
| 9. cruel | 7 | (9, 1-4-7)$_0^1$ (9, 2-3-5-6-8)$_0^2$ |
| 10. relentless | 8 | (10, 9)$_0^1$ (10, 1-4-7)$_0^2$ (10, 2-3-5-6-8)$_0^5$ |
| 11. weird | 8 | (11, 3)$_0^1$ (11, 1-4-7-10)$_0^2$ (11, 2-3-5-6-8)$_0^5$ |
| 12. eerie | 9 | (12, 3-11)$_0^1$ (12, 1-4-7-10)$_0^2$ (12, 2-3-5-6-8)$_0^5$ |
| 13. cold | 11 | (13, 3-6-7-8-11-12)$_0^1$ (13, 1-4-9)$_0^2$ (13, 2-3-5-6-10)$_0^5$ |
| 14. barren | 11 | (14, 6-7)$_0^2$ (14, 1-2-3-4-5-8-9-10-11-12-13)$_1^{T7}$ |
| 15. sterile | 11 | (15, 14)$_0^1$ (15, 6-7)$_0^2$ (15, 1-2-3-4-5-8-9-10-11-12-13)$_1^{T7}$ |
| 16. hated | 12 | (16, 7)$_0^0$ (16, 1-4-6-8-9-13)$_0^1$ (16, 14-15)$_0^2$ (16, 2-3-5-10-11-12)$_0^5$ |
| 17. cruel | 12 | (17, 9)$_0^0$ (17, 1-4-7-10)$_0^1$ (17, 2-3-5-6-8-11-12-13)$_0^5$ (17, 14-15)$_1^{T7}$ |

| Chain 2, Segment 2 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 18. perversely | 16 | $(18, 10)_0^2$ $(18, 1\text{-}2\text{-}3\text{-}4\text{-}5\text{-}6\text{-}7\text{-}8\text{-}9\text{-}11\text{-}12\text{-}13\text{-}16\text{-}17)_1^{T10}$ |

| Chain 2, Segment 3 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 19. cruel | 24 | $(19, 9\text{-}17)_0^0$ $(19, 1\text{-}4\text{-}7\text{-}10)_0^1$ $(19, 2\text{-}3\text{-}5\text{-}6\text{-}8\text{-}11\text{-}12\text{-}13)_0^5$ $(19, 14\text{-}15)_1^{T7}$ |

| Chain 3 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. married | 13 | |
| 2. wife | 14 | $(2, 1)_0^1$ |
| 3. wife | 15 | $(3, 1)_0^1$ $(3, 2)_0^0$ |
| 4. wife | 27 | $(4, 2\text{-}3)_0^0$ $(4, 1)_0^1$ |

| Chain 4 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. conceded | 19 | |
| 2. tolerance | 20 | $(2, 1)_0^1$ |

| Chain 5 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. virgin | 31 | |
| 2. pine | 31 | |
| 3. bush | 31 | $(3, 1)_0^1$ |
| 4. trees | 32 | $(4, 1)_0^1$ $(4, 3)_0^1$ |
| 5. trunks | 32 | |
| 6. trees | 33 | $(6, 4)_0^0$ $(6, 1\text{-}3)_0^1$ |

| Chain 6 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. hand-in-hand | 34 | |
| 2. matching | 34 | |
| 3. whispering | 35 | |
| 4. laughing | 35 | |
| 5. warm | 38 | $(5, 1)_0^1$ $(5, 4)_0^5$ |

| Chain 7 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. first | 1 | |
| 2. initial | 1 | $(2, 1)_0^1$ |
| 3. final | 2 | $(3, 2\text{-}1)_0^3$ |

| Chain 8 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. night | 2 | |
| 2. dusk | 3 | $(2, 1)_0^2$ |
| 3. darkness | 3 | $(3, 1\text{-}2)_0^1$ |

| Chain 9 | | |
|---|---|---|
| Word | Sentence | Lexical Chain |
| 1. environment | 7 | |
| 2. setting | 7 | |
| 3. surrounding | 8 | |