WordNet-Based Inference of Textual Cohesion and Coherence

Sanda M. Harabagiu
Artificial Intelligence Center
SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025
harabagi@ai.sri.com

Abstract

This paper[1] presents a computational method for the recognition of the cohesive and coherence structures of texts. A large lexical knowledge base built on top of WordNet provides with the lexico-semantic information that needs to be mined. A path-finding algorithm returns the cohesive structure of a text with results that outperform previous approaches.

The lexical paths contained in the cohesive structures are used to (1) build patterns of association between cue phrases and coherence relations and (2) to find the lexical characteristics of coherence categories. Finally, the textual coherence structure is recognized by giving priority to the coherence constrains induced by cue phrases. The paper presents also the performance of building the coherence structure for several texts.

Introduction

In a text, a sequence of sentences tends to convey information about a certain topic, and by doing so, they use related words, providing the text with the quality of unity. This property of sentences of "sticking together" to function as a whole, as defined in (Halliday and Hassan 1976) and (Morris and Hirst 1991) is known as *cohesion*. A sequence of sentences in a text must also display logical connections, accounting for the *coherence* of the text. If cohesion is a term for sticking together, then coherence is a term for making sense.

In this paper, we revisit the notion of lexical cohesion, and present its contribution to the evaluation of text coherence. We make use of the vast lexical knowledge rendered by WordNet (Miller 1995) to build lexical paths spanning the words of texts. Lexical cohesion, resulting from novel techniques of searching the WordNet thesaurus, is shown to contribute to an automatic approach of discourse coherence analysis. We develop a methodology of discovering coherence patterns from the lexical cohesion of texts, using an initial set of coherence rules inspired by the Hovy's taxonomy of discourse relations (Hovy 1993). Its performance is compared to Marcu's recent Rhetorical Parser(Marcu 1997).

The WordNet lexical database

WordNet (Miller 1995) is a machine-readable dictionary designed at Princeton, following psycholinguistic principles. Unlike standard alphabetical dictionaries which organize vocabularies using morphological similarities, WordNet organizes lexical information in terms of word meanings. WordNet encodes 91,595 sets of synononym words (know as *synsets*), covering the large majority of English nouns, verbs, adjectives and adverbs. Words having multiple semantic meanings belong to as many synsets as their meanings, which are ordered along their frequency of occurance in real texts.

Words and their underlying concepts are linked in WordNet through thirteen types of lexico-semantic relations, presented in (Miller 1995). Most of the 391,885 relations encoded in WordNet are represented by *is-a* relations that create hierarchies of nouns and verbs. Some meronym (*is_part*, *is_member*, *has_stuff*) relations between noun concepts are also represented. Additionally, verbs are connected through two kind of semantic relations inspired by logic implications: *entail* and *cause_to*. There are also relations that are induced by derivational morphology: *pertaynym* and *attribute*.

The existing semantic network can be enriched with more relations, inspired by the typical thematic roles encountered in real world texts. Some of these relations are: *agent*, *object*, *instrument*, *beneficiary*, *location*, *state*, *reason*, *theme* or *manner*. Such relations were acquired from the corpus of gloss definitions provided by WordNet, and called *gloss relations*. This is part of the methodology of mapping the gloss definitions into semantic networks, presented in (Harabagiu

1997).

A path-finding algorithm

A path-finding algorithm was designed to find semantic paths between the words of a text. It consists of four steps. that successively perform searches in WordNet and consolidate the cohesion structures. The search mechanism uses three types of primitives that construct a semantic path between a pair of concepts by using a different set of knowledge base relations. These primitives establish between a pair of WordNet concepts (C_1, C_2) : (i) *simple connections* when there is a concept C_3 such that there is a sequence of WordNet relations r_s from C_1 to C_3 and another one from C_2 to C_3: (ii) *gloss connections* if there is a sequence of gloss relations r_g from C_1 to a gloss concept C_3 and another one from C_2 to C_3: and (iii) *combined connections* if there is a WordNet or a gloss concept C_3 such that it is connected to C_1 and C_2 by sequences of WordNet or gloss relations. These primitives are used over and over in all the four steps of the path-finding algorithm:

Step 1: Find paths that explain textual relations

For a given lexical relation r that links $concept_i$ to $concept_j$. the search strategy is to look for the same lexical relation r in the glosses of nearby concepts. We have investigated three methods that have different search strategies. The first method searches for relation r first in the gloss of $concept_i$ and then in the glosses of concepts that connect with $concept_i$ via simple/gloss/combined paths. When relation r is found in such a gloss, connections are sought between the destination concept of that relation and $concept_j$.

The second method is used when r is adjacent in the text to a relation r_1. This method searches for relation r in the glosses of concepts linked via a simple/gloss/combine connection to the address concept of r_1. When relation r is found in such a gloss, connections are sought between the destination concept of that relation and $concept_j$.

Finally, the third method, searches first for all glosses that contain $concept_i$ and marks the concepts in these glosses as $gloss\_concept_i$. Then, the method searches for relation r in the gloss of some concept that connects to any of the $gloss\_concept_i$ identified before. When relation r is found, simple/gloss/combined connections are sought between its destination concept and $concept_j$.

Step2: Determine the local context of a sentence

The role of this step is to merge the paths found in Step 1 for all lexical relations of a sentence into a graph where common concepts are not repeated. This consolidated graph is considered to represent the context of that sentence. The result of this step is a web of concepts connected through relations that were building the paths detected at step 1.

Step3: Find cohesion paths between sentences

This step takes advantage of the collective meaning of all sentences in the text by finding connections between the local contexts. We have developed three different ways of retrieving connections between the concepts of different sentences. One way is to find knowledge base paths between the *verbs* of one sentence and the *verbs* of the other sentence. We select only verbs since they induce the dominant knowledge of a sentence: knowledge about actions, states or events. The second method finds paths between the verbs of one sentence and the nodes of the local context of the other sentence. A third way is to pair the nodes from the local contexts of one sentence with the nodes from the local context of the other sentence. For each sentence, its connections to all previously processed sentences are searched.

Step4: Consolidate the cohesion structure of a text

The global cohesion structure of a sequence of sentences is achieved by eliminating the repeating concepts throughout textual paths and local contexts. First, the common concepts between the textual connections are reduced by applying the same procedure as the one used in Step 2 for building the local contexts. Then, this new structure is matched against each local context, and common concepts are further reduced.

Lexical paths as forms of cohesion

The first algorithm that searched for lexical cohesion relations in texts was devised by Morris and Hirst. Their approach found well over 90% of the intuitive lexical relations from a set of five examples presented in (Morris 1988). and was able to retrieve 14 out of the 16 nonsystematic lexical chains given as examples in (Halliday and Hassan 1976) (thus an 87% recall). These promising results prompted the consideration of using WordNet for the detection of lexical cohesion relations from the large corpus provided by Treebank (Marcus et. al.1993). In the process. we discovered interesting associations with the approach of Morris and Hirst, as well as complex divergences.

The algorithm devised by Morris and Hirst to build lexical chains uses five types of thesaural relations that can be generalized to the simple. gloss or combined connections used by the path-finding algorithm. In contrast. the path-finding algorithm provides with a wealth of lexical cohesion relations. most of them uncovered by the algorithm of Morris and Hirst. For example. for the text presented in (Morris and Hirst 1991), we found 38 lexical paths as opposed to their 9 lexical chains. Our results, fully detailed in (Harabagiu 1997) show an increase in the recall with 44%. The precision is enforced as well, since the paths have to comply with the constraints of the local contexts. Even for the paths that correspond to their lexical chains, the inter-relationships between the words were more dense.

Cue Phrases as Coherence Indicators

Discourse cue phrases are words and phrases that signal information regarding the logical flow of the discourse, e.g. the coherence relations among discourse fragments. However, the majority of the cue phrases are ambiguous, in the sense that they have also alternative meanings, where the word doesn't contribute to the discourse level semantics, but rather to the semantic meaning of the sentences.

Building on the previous work encompassing the studies presented in (Hirshberg and Litman 1993), (Siegel and McKeown 1994), (Grosz and Sidner 1996), the approach used in (Marcu 1997), extends the problem of cue phrase disambiguation by distinguishing the discourse sense of a cue phrase into finer meanings, corresponding to the rhetorical relations it indicates. Our approach has many similarities with Marcu's method because we focus on the recognition of a basic set of relations derived from the top of the taxonomy obtained by Hovy in (Hovy 1993).

The cardinality of the set of potential discourse markers we considered is far smaller than the one used by Marcu. We have been considering only 29 cue phrases, as opposed to Marcu's study of 450 discourse markers. The difference in size may be motivated by the fact that we aimed at complex lexico-semantic processing of each example, and thus required more effort per cue phrase.

Since our focus is on the correspondence between cue phrases, semantic paths and coherence relations, we gathered all the paths tagged with the same cue phrase and the same coherence relation in classes $C^{cue-phrase}_{coherence-relation}$. Next each path from every $C^{cue-phrase}_{coherence-relation}$ is transformed into a pattern by applying the following succession of operations:

o 1. Every synset is replaced with its part-of-speech tag. Therefore, every concept is represented only by its syntactic category.

o 2. Successions of the same relation in a path are substituted by an instance of that relation, connecting the first and the last argument of the chain.

o 3. Every succession of gloss relations is replaced by a single relation, connecting the first gloss concept to the last gloss concept from path. This new relation is associated with a list, containing all the labels of the relations it substitutes in the original path.

o 4. Pattern extraction is performed, by identifying the longest subpath that is common to most of the members of the class. Patterns are formed as regular expressions (of part-of-speech tags and directed relation labels) in which the common subpath is identical, whereas the disjoint parts gather all the substitutable relations that can be found in the various transformed paths of that class.

The evaluation of the cue-phrase disambiguation approach was performed on two different sets of texts, pertaining to different genres: a collection of Wall Street Journal articles, using 1403 words, and a 1528 word long fragment from the scientific abstracts provided by the U.S. Department of Energy, both available from the Treebank project. Three independent judges identified the meanings of the cue phrases and validated the results of the disambiguation procedure. The results show that 86.45% of the discourse senses of the cue phrases were discovered with a precision of 72.91%, a result which is close to what Marcu obtained (Marcu 1997) with a surface-based algorithm.

Text Coherence

It is well established that the structure of a text contains more than the collection of the sentence structures: its meaning is determined by the logical relations between sentences. This additional meaning is provided by the inferences establishing the interpretation of the text under the assumption that it is coherent. Coherence inference relies on pragmatic knowledge, using various aspects of commonsense reasoning mechanisms. Here, we describe the effect of knowledge gathered from a large linguistic database on the recognition of coherence relations and on the general structure of the discourse.

We consider a taxonomy of coherence relations, initially reported in (Maier and Hovy 1992) that is mapped into the coherence categories devised in (Kehler 1995). These coherence categories are characterized by properties that can be recognized from the information brought forward by the lexico-semantic paths. Lists of cohesive constraints, as indicated by properties of the sequences derived from the WordNet paths, are derived and help recognize each coherence relation.

The defining constraints of the coherence relations also determine the text spans underlying the coherence structure of a text. *Resemblance* or *Cause-Effect* relations can be recognized between pairs of textual units (clauses or sentences), whereas *Contiguity* relations organize the other binary relations into segments of coherence structures. We favor this organization of the textual coherence structure to the hierarchi-

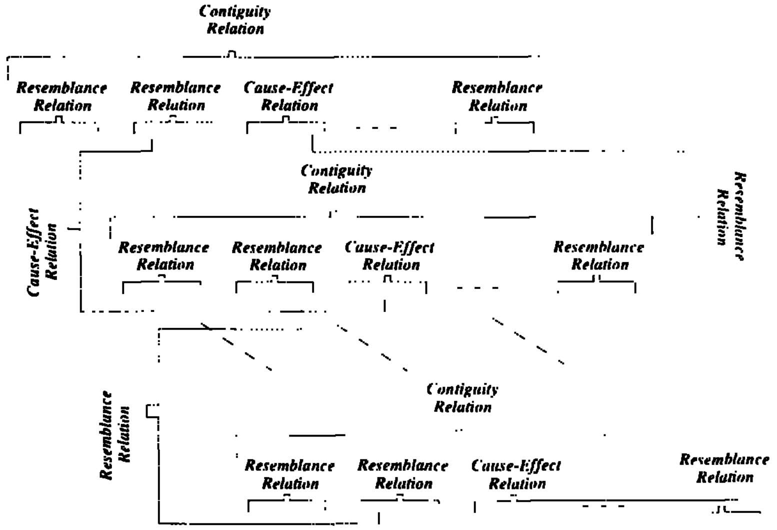

Figure 1: Coherence structure based on semantic paths

cal organization. The only other automatic method of building the rhetorical tree of a text we are aware of is the method implemented in (Marcu 1996). Marcu reformulates the definition of the structure of a text as devised by the RST (Mann and Thompson 1988) relations. He considers a formal treatment of the coherence structure by relying on a shallow discourse analyzer based on cue phrase information.

The inspection of the coherence and cohesion constraints imposed by the structure of semantic paths triggers the discovery of *Resemblance* and *Cause-Effect* relations. The *Contiguity* relations are determined by constraints that involve *Resemblance* or *Cause-Effect* relations, producing a higher level of the text coherence structure. Figure 1 illustrates a possible text coherence structure.

The fact that coherence constraints use information rendered by the cohesion paths permits the detection of coherence even when cue phrases are not present in a text passage is the main departure we take from Marcu's approach. The coherence structure of a text is produced by the steps of the algorithm:

```
Algorithm build_coherence_structure (text )
1. Build the semantic paths spanning the text;
2. Recognize discourse cue phrases and their
   corresponding coherence relation;
3. Find resemblance and cause-effect relation between
   text units connected by more than 4 semantic paths.
4. if (the cue phrase indicates another coherence relation)
       then select that relation;
5. Find contiguity relations spanning text passages
   covered by a dense webs of semantic paths;
6. Output the coherence structure:
     (coherence relation; text passage: semantic paths);
```

Discussion and evaluation

The performance evaluation of the algorithm for finding the coherence structure of a text was done by considering more than 10 texts from the *Wall Street Journal* corpus available in the Treebank project. We have then grouped the texts according to their length (i.e. number of sentences) and analyzed the characteristics of their coherence structures.

The results indicate that there are about 35% more coherence relations than the number of sentences in a text and that less than 30% of these relations are signaled by cue phrases. This indicates a "lighter" coherence structure than that rendered by Marcu's rhetorical trees, which for a text of n units (and thus $n/2$ coherence relations between the pairs of textual units) builds a binary tree with $2^{n/2+1}$ rhetorical relations. Almost half of the coherence relations are resemblance relations and the number of contiguity relations varies slightly. The same measurements performed on texts of different size indicate that in fact, the number of contiguity relations depends on the size of the text.

For each of the texts upon which the algorithm has built the coherence structures, three analysts constructed manually the discourse structure, given access to the semantic paths returned by the path-finding algorithm for the texts. Then, separately, each analyst was also given first information regarding the coherence relation signaled by cue phrases and then the semantic patterns derived from the paths, signaling coherence relations. Whenever at least two of the humans tagged a text passage with the same coherence relation as the the one in the automatic structure, we considered a hit, in other cases a miss. Table 1 illustrates the precision and correctness obtained for five texts from Treebank (Marcus et al.1993). The number n_1 stands for the number of coherence relations identified manually. n_2 represents the number of relations identified by the algorithm and n_3 is the number of coherence relations correctly identified.

Text	n_1	n_2	n_3	Recall	Precision
w0741.par	40	51	25	78.43%	49.01%
w0745.par	33	39	19	84.61%	48.71%
w0748.par	35	43	21	81.39%	48.83%
w0764.par	58	72	33	80.55%	45.83%
w0778.par	32	44	20	72.72%	45.45%

Table 1: Evaluation of the coherence-structure building algorithm

The algorithm found around 80% of the coherence relations, but the correctness of the relations is below 50%. The low precision has the explanation that it was difficult to find agreement between the judgments of the humans and the output of the algorithm. One of the possible motivation for this may be the fact that we considered an insufficient number of coherence relations. The inclusion of more relations from Hovy's taxonomy and the analysis of their dependence on semantic paths may increase the precision of the algorithm. Nevertheless, the values of the *recall* are acceptable, showing that most of the coherence structure of the texts was discovered.

It would be interesting to measure the relevance and correctness of the coherence structure returned by this algorithm against a corpus of discourse structures, but unfortunately such resources are not yet available in the computational linguistics community. The only automatic coherence builder for English we are aware of is Marcu's Rhetorical parser, therefore we assessed the correctness of our algorithm by measuring the agreement with the rhetorical structure built by Marcu for the same text. The experiments are detailed in (Harabagiu 1997) and show that we obtained almost 80% identical coherence structures. The knowledge inferred by the coherence structures of texts was also used for solving coreference in texts and ported significant improvements in precision, fully detailed in (Harabagiu 1997).

References

B.J. Grosz and C.L. Sidner. Attention, intentions and the structure of discourse. *Computational Linguistics*, 12(2):175-204, 1986.

M.A.K. Halliday and R. Hasan. *Cohesion in English*. Longman, London, 1976.

S.M. Harabagiu. *WordNet-based inference of textual context, cohesion and coherence*. PhD thesis, University of Southern California, Los Angeles, CA, 1997.

J. Hirshberg and D Litman. Empirical studies on the disambiguation of cue phrases. *Computational Linguistics*, 19(3):501-530, 1993.

E.H. Hovy. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63:341-385, 1993.

A. Kehler. *Interpreting Cohesion Forms in the Context of Discourse Inference*. PhD thesis, Harvard University, Cambridge, MA, 1995.

E. Maier and E. Hovy. Organizing discourse structure relations using metafunctions. In H. Horacek, editor, *New Concepts in Natural Language Generation: Planning, Realization and Systems*, pages 178-201. Pinter, London, 1992.

W.C. Mann and S. Thompson. Rhetorical structure theory. *Text*, 8:243-281, 1988.

D. Marcu. Building up rhethorical structure trees. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-96)*, pages 1069-1074, Portland, OR, 1996.

D. Marcu. The rhethorical parsing of natural language texts. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, 1997.

M. Marcus, B. Santorini and M.A. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313-330, 1993.

G.A. Miller. WordNet: A Lexical Database. *Communication of the ACM*, vol 38: No11, pages 39-41, November 1995.

J. Morris. Lexical cohesion, the thesaurus, and the structure of text. Master's thesis, University of Toronto, Toronto, Canada, 1988.

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

E.V. Siegel and K.R. McKeown. Emergent linguistic rules from inducing decision trees: Disambiguating discourse clue words. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-96)*, pages 820-826, Seattle, WA, 1994.