An effective Discourse Parser that uses Rich Linguistic Information

Rajen Subba *
Display Advertising Sciences
Yahoo! Labs
Sunnyvale, CA, USA
rajen@yahoo-inc.com

Barbara Di Eugenio
Department of Computer Science
University of Illinois
Chicago, IL, USA
bdieugen@cs.uic.edu

Abstract

This paper presents a first-order logic learning approach to determine rhetorical relations between discourse segments. Beyond linguistic cues and lexical information, our approach exploits compositional semantics and segment discourse structure data. We report a statistically significant improvement in classifying relations over attribute-value learning paradigms such as Decision Trees, RIPPER and Naive Bayes. For discourse parsing, our modified shift-reduce parsing model that uses our relation classifier significantly outperforms a right-branching majority-class baseline.

1 Introduction

Many theories postulate a hierarchical structure for discourse (Mann and Thompson, 1988; Moser et. al., 1996; Polanyi et. al., 2004). Discourse structure is most often based on semantic / pragmatic relationships between spans of text and results in a tree structure, as that shown in Figure 1. Discourse parsing, namely, deriving such tree structures and the *rhetorical relations* labeling their inner nodes is still a challenging and mostly unsolved problem in NLP. It is linguistically plausible that such structures are determined at least in part on the basis of the meaning of the related chunks of texts, and of the rhetorical intentions of their authors. However, such knowledge is extremely difficult to capture. Hence, previous work on discourse parsing (Wellner et. al., 2006; Sporleder and Lascarides, 2005; Marcu, 2000; Polanyi et. al., 2004; Soricut and Marcu, 2003; Baldridge and Lascarides, 2005) has relied only on syntactic and lexical information, lexical chains and shallow semantics.

We present an innovative discourse parser that uses compositional semantics (when available) and information on the structure of the segment being built itself. Our discourse parser, based on a modified shift-reduce algorithm, crucially uses a rhetorical relation classifier to determine the site of attachment of a new incoming chunk together with the appropriate relation label. Another novel aspect of our work is the usage of Inductive Logic Programming (ILP): ILP learns from first-order logic representations (FOL). The ILP-based relation classifier is significantly more accurate than relation classifiers that use competitive propositional ML algorithms such as decision trees and Naive Bayes. In addition, it results in FOL rules that are linguistically perspicuous. Our domain is that of instructional how-to-do manuals, and we describe our corpus in Section 2. In Section 3, we discuss the modified shift-reduce parser we developed. The bulk of the paper is devoted to the rhetorical relation classifier in Section 4. Experimental results of both the relation classifier and the discourse parser in its entirety are discussed in Section 5. Further details can be found in (Subba, 2008).

2 Discourse Annotated Instructional Corpus

Existing corpora annotated with rhetorical relations (Carlson et. al., 2003; Wolf and Gibson, 2005; Prasad et. al., 2008) focus primarily on news articles. However, for us the development of the discourse parser is parasitic on our ultimate goal: developing resources and algorithms for language in-

*This work was done while the author was a student at the University of Illinois at Chicago.

Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pages 566–574,
Boulder, Colorado, June 2009.

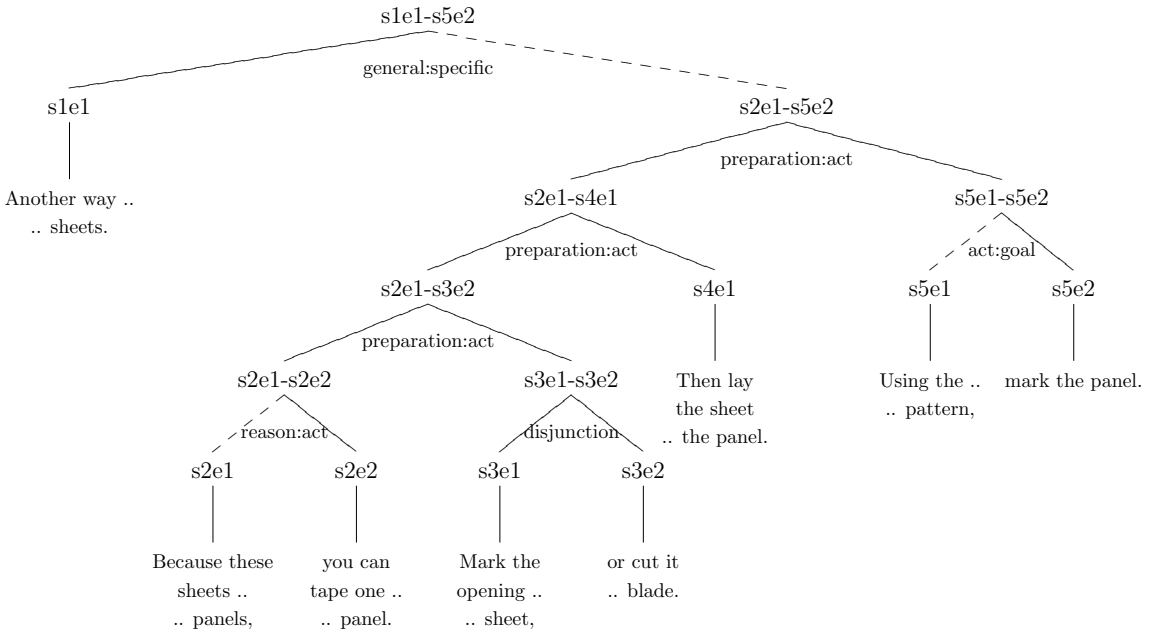


Figure 1: Discourse Parse Tree of the Text in Example (1)

terfaces to instructional applications. Hence, we are interested in working with instructional texts. We worked with a corpus on home-repair that is about 5MB in size and is made up entirely of written English instructions,[1] such as that shown in Example (1). The text has been manually segmented into Elementary Discourse Units (EDUs), the smallest units of discourse. In total, our corpus contains 176 documents with an average of 32.6 EDUs for a total of 5744 EDUs and 53,250 words. The structure for Example (1) is shown in Figure 1.

(1) [Another way to measure and mark panels for cutting is to make a template from the protective sheets.$_{(s1e1)}$] [Because these sheets are the same size as the panels,$_{(s2e1)}$] [you can tape one to the wall as though it were a panel.$_{(s2e2)}$] [Mark the opening on the sheet$_{(s3e1)}$] [or cut it out with a razor blade.$_{(s3e2)}$] [Then lay the sheet on the panel.$_{(s4e1)}$] [Using the template as a pattern,$_{(s5e1)}$] [mark the panel.$_{(s5e2)}$]

To explore our hypothesis, that rich linguistic information helps discourse parsing, and that the state of the art in machine learning supports such an approach, we needed training data annotated with both compositional semantics and rhetorical relations. We performed the first type of annotation almost completely automatically, and the second manually, as we turn now to describing.

2.1 Compositional Semantics Derivation

The type of compositional semantics we are interested in is heavily rooted in verb semantics, which is particularly appropriate for the instructional text we are working with. Therefore, we used VerbNet (Kipper et. al., 2000) as our verb lexicon. VerbNet groups together verbs that undergo the same syntactic alternations and share similar semantics. It accounts for 4962 distinct verbs classified into 237 main classes. Each verb class is described by thematic roles, selectional restrictions on the arguments and frames consisting of a syntactic description and semantic predicates. Such semantic classification of verbs can be helpful in making generalizations, especially when data is not abundant. Generalization can also be achieved by means of the semantic predicates. Although the verb classes of two verb instances may differ, semantic predicates are shared across verbs. To compositionally build verb based

[1]The raw corpus was originally assembled at the Information Technology Research Institute, University of Brighton.

semantic representations of our EDUs, we (Subba et al., 2006) integrated a robust parser, LCFLEX (Rosé, 2000), with a lexicon and ontology based both on VerbNet and, for nouns, on CoreLex (Buitelaar, 1998). The augmented parser was able to derive complete semantic representations for 3257 of the 5744 EDUs (56.7%). The only manual step was to pick the correct parse from a forest of parse trees, since the output of the parser can be ambiguous.

2.2 Rhetorical relation annotation

The discourse processing community has not yet reached agreement on an inventory of rhetorical relations. Among the many choices, our coding scheme is a hybrid of (Moser et. al., 1996) and (Marcu, 1999). We focused on what we call *informational relations*, namely, relations in the domain. We used 26 relations, divided into 5 broad classes: 12 **causal** relations (e.g., *preparation:act, goal:act, cause:effect, step1:step2*); 6 **elaboration** relations (e.g., *general:specific, set:member, object:attribute*; 3 **similarity** relations (*contrast1:contrast2, comparison, restatement*); 2 **temporal** relations (*co-temp1:co-temp2, before:after*); and 4 **other** relations, including *joint* and *disjunction*.

The annotation yielded 5172 relations, with reasonable intercoder agreement. On 26% of the data, we obtained $\kappa = 0.66$; κ rises to 0.78 when the two most commonly confused relations, *preparation:act* and *step1:step2*, are consolidated. We also annotated the relata as *nucleus* (more important member) and *satellite* (contributing member(s)) (Mann and Thompson, 1988), with $\kappa = 0.67$.[2] The most frequent relation is *preparation:act* (24.46%), and in general, causal relations are more frequently used in our instructional corpus than in news corpora (Carlson et. al., 2003; Wolf and Gibson, 2005).

3 Shift-Reduce Discourse Parsing

Our discourse parser is a modified version of a shift-reduce parser. The shift operation places the next segment on top of the stack, TOP. The reduce operation will attach the text segment at TOP to the text segment at TOP-1. (Marcu, 2000) also uses a shift-reduce parser, though our parsing algorithm differs in two respects: 1) we do not learn shift operations and 2) in contrast to (Marcu, 2000), the attachment of an incoming text segment to the emerging tree may occur at any node on the right frontier. This allows for the more sophisticated type of adjunction operations required for discourse parsing as modeled in D-LTAG (Webber, 2004). A reduce operation is determined by the relation identification component. We check if a relation exists between the incoming text segment and the attachment points on the right frontier. If more than one attachment site exists, then the attachment site for which the rule with the highest score fired (see below) is chosen for the *reduce* operation. A reduce operation can further trigger additional reduce operations if there is more than one tree left in the stack after the first reduce operation. When no rules fire, a *shift* occurs. In the event that all the segments in the input list have been processed and a full DPT has not been obtained, then we reduce TOP and TOP-1 using the *joint* relation until a single DPT is built.

4 Classifying Rhetorical Relations

Identifying the informational relations between text segments is central to our approach for building the informational tree structure of text. We believe that the use of a limited knowledge representation formalism, essentially propositional logic, is not adequate and that a relational model that can handle compositional semantics is necessary. We cast the problem of determining informational relations as a classification task. We used the ILP system Aleph that is based on (Muggleton, 1995). Formulation of any problem within the ILP framework consists of background knowledge **B** and the set of examples **E** ($E^+ \cup E^-$). In our ILP framework, positive examples are ground clauses describing a relation and its relata, e.g. *relation(s5e1,s5e2,act:goal)*, or *relation(s2e1-s3e2,s4e1,preparation:act)* from Figure 1. If e is a positive example of a relation r, then it is also a negative example for all the other relations.

Background Knowledge (**B**) can be thought of as features used by ILP to learn rules, as in traditional attribute-value learning algorithms. We use the following information to learn rules for classifying relations. Figure 2 shows a sample of the background

[2] We don't have space to explain why we annotate for nucleus and satellite, even if (Moser et. al., 1996) argue that this sort of distinction does not apply to informational relations.

Verbs + Nouns:	verb('$s5e2$',mark). noun('$s5e2$',panel).
Linguistic Cues:	firstWordPOS('$s5e2$','VB'). lastWordPOS('$s5e2$','.').
Similarity:	segment_ sim_ score('$s5e1$','$s5e2$',0.0).
Compositional Semantics:	verbclass('$s5e2$',mark,'image_impression-25.1'). agent('$s5e2$',frame(mark),you). destination('$s5e2$',frame(mark),panel). cause('$s5e2$',frame(mark),you,'$s5e2$-mark-e'). prep('$s5e2$',frame(mark),end('$s5e2$-mark-e'),mark,panel). created_image('$s5e2$',frame(mark),result('$s5e2$-mark-e'),mark).
Structural Information:	same_sentence('$s5e1$','$s5e2$').

Figure 2: Example Background Knowledge

knowledge provided for EDU $s5e2$.

Verbs + Nouns: These features were derived by tagging all the sentences in the corpus with a POS tagger (Brill, 1995).

WordNet: For each noun in our data, we also use information on hypernymy and meronymy relations using WordNet. In a sense, this captures the domain relations between objects in our data.

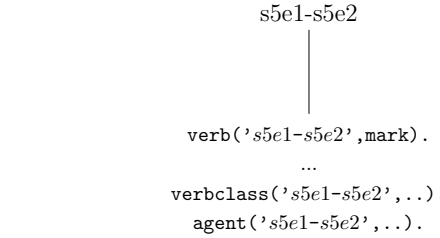
Linguistic Cues: Various cues can facilitate the inference of informational relations, even if it is well known that they are based solely on the content of the text segments, various cues can facilitate the inference of such relations. At the same time, it is well known that relations are often non signalled: in our corpus, only 43% of relations are signalled, consistently with figures from the literature (44% in (Williams and Reiter, 2003) and 45% in (Prasad et. al., 2008)). Besides lexical cues such as *but*, *and* and *if*, we also include modals, tense, comparatives and superlatives, and negation. E.g., *wrong-act* in relations like *prescribe-act:wrong-act* is often expressed using a negation.

Similarity: For the two segments in question, we compute the cosine similarity of the segments using only nouns and verbs.

Compositional semantics: the semantic information derived by our parser, as described in Section 2.1. The semantic representation of segment $s5e2$ from Example (1) is shown in Figure 2. Each semantic predicate is a feature for the classifier.

Structural Information: For relations between two EDUs, we use knowledge of whether the two EDUs are intra-sentential or inter-sentential, since some relations, e.g. *criterion:act*, are more likely to be realized intra-sententially than inter-sententially. For larger segments, we also encode the hierarchical representation of text segments that contain more than one nucleus, the distance between the nuclei of the two segments and any relations that exist between the smaller inner segments.

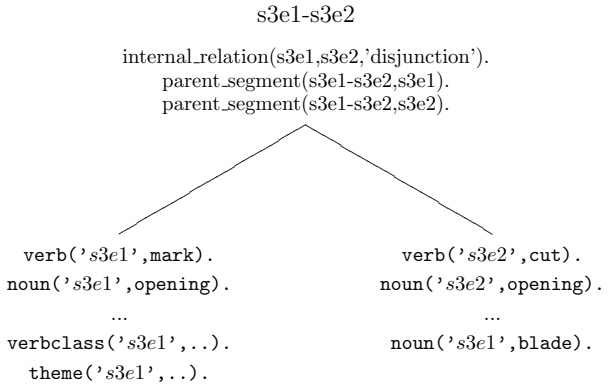
At this point, the attentive reader will be wondering how we encode compositional semantics for relations relating text segments larger than one EDU. Clearly we cannot just list the semantics of each EDU that is dominated by the larger segment. We follow the intuition that nuclei represent the most important portions of segments (Mann and Thompson, 1988). For segments such as $s5e1$-$s5e2$ that contains a single nucleus, we simply reduce the semantic content of the larger segment to that of its nucleus:



In this case, the semantics of the complex text segment is represented by the compositional semantics of the single most important EDU.

For segments that contain more than one nucleus, such as $s3e1$-$s3e2$, the discourse structure information of the segment is represented with the additional predicates *internal_relation* and *parent_segment*. These predicates can be used recursively at every level of the tree to specify the relation between the most important segments. In addition, they also provide a means to represent the compositional semantics of the most important EDUs and

make them available to the relational learning algorithm.



4.1 Learning FOL Rules for Discourse Parsing

In Aleph, the hypothesis space is restricted to a set of rules that conform to a predefined language *L*. This is done with the use of *mode declarations* which, in other words, introduces a language bias in the learning process. *modeh* declarations inform the learning algorithm about what predicates to use as the head of the rule and *modeb* specifies what predicates to use in the body of the rule. Not all the information in **B** needs to be included in the body of the rule. This makes sense since we often learn definitions of concepts based on more abstract higher level information that is inferred from some other information that is not part of our final definition. Mode declarations are used by Aleph to build the most specific clause ($\perp$) that can be learned for each example. $\perp$ constrains the search for suitable hypotheses. $\perp_i$ is built by taking an example $e_i \in \boldsymbol{E}^+$ and adding literals that are entailed by **B** and e_i. We then have the following property, where H_i is the hypothesis (rule) we are trying to learn and $\preceq$ is a generality operator:

$$\square \preceq H_i \preceq \perp_i$$

Finding the most specific clause ($\perp$) provides us with a partially ordered set of clauses from which to choose the best hypothesis based on some quantifiable qualitative criteria. This sub-lattice is bounded by the most general clause ($\square$, the empty clause) from the top and the most specific clause ($\perp$) at the bottom. We use the heuristic search in Aleph that is similar to the A*-like search strategy presented by (Muggleton, 1995) to find the best hypothesis (rule). A noise threshold on the number of negative examples that can be covered by a rule can be set. We learn a model that learns perfect rules first and then one that allows for at most 5 negative examples. A backoff model that first uses the model trained with $noise = 0$ and then $noise = 5$ if no classification has been made is used. We use the evaluation function in Equation 1 to guide our search through the tree of possible hypotheses. This evaluation function is also called the compression function since it prefers simpler explanations to more complex ones (Occam's Razor). f_s is the score for clause c_s that is being evaluated, p_s is the number of positive examples, n_s is the number of negative examples, l_s is the length of the clause (measured by the number of clauses).

$$f_s = p_s - (n_s + (0.1 \times l_s)) \quad (1)$$

Classification in most ILP systems, including Aleph, is restricted to binary classification (positive vs. negative). In many applications with just two classes, this is sufficient. However, we are faced with a multi-classification problem. In order to perform multi-class classification, we use a decision list. First, we build *m* binary classifiers for each relation $r \in R$. Then, we form an ordered list of the rules based on the following criterion:

1. Given two rules r_i and r_j, r_i ,is ranked higher than r_j if $(\mathrm{p}_i - n_i) > (\mathrm{p}_j - n_j)$.
2. if $(\mathrm{p}_i - n_i) = (\mathrm{p}_j - n_j)$, then r_i is ranked higher than r_j if $(\frac{p_i}{p_i+n_i}) > (\frac{p_j}{p_j+n_j})$.
3. if $(\mathrm{p}_i - n_i) = (\mathrm{p}_j - n_j)$ and $(\frac{p_i}{p_i+n_i}) = (\frac{p_j}{p_j+n_j})$ then r_i is ranked higher than r_j if $(\mathrm{l}_i) > (\mathrm{l}_j)$.
4. default: random order

Classifying an unseen example is done by using the first rule in the ordered list that satisfies it.

5 Experiments and Results

We report our results from experiments on both the classification task and the discourse parsing task.

5.1 Relation Classification Results

For the classification task, we conducted experiments using the stratified k-fold (k = 5) cross-validation evaluation technique on our data. Unlike

(Wellner et. al., 2006; Sporleder and Lascarides, 2005), we do not assume that we know the order of the relation in question. Instead we treat reversals of non-commutative relations (e.g. *preparation:act* and *act:goal*) as separate relations as well. We compare our ILP model to RIPPER, Naive Bayes and the Decision Tree algorithm. We should point out that since attribute-value learning models cannot handle first-order logic data, they have been presented with features that lose at least some of this information. While this may then seem to result in an unfair comparison, to the contrary, this is precisely the point: can we do better than very effective attribute-value approaches that however inherently cannot take richer information into account? All the statistical significance tests were performed using the value of F-Score obtained from each of the folds. We report performance on two sets of data since we were not able to obtain compositional semantic data for all the EDUs in our corpus:

- Set A: Examples for which semantic data was available for all the nuclei of the segments (1789 total). This allows us to have a better idea of how much impact semantic data has on the performance, if any.
- Set B: All examples regardless of whether or not semantic data was available for the nuclei of the segments (5475 total).

Model	Semantics	No Semantics
ILP	**62.78**	60.25
Decision Tree	56.29	55.45
RIPPER	58.02	56.96
Naive Bayes	35.83	34.66
Majority Class	31.63	31.63

Table 1: Classification Performance: Set A (F-Score)

Table 1 shows the results on Set A. ILP outperforms all the other models. Via ANOVA, we first conclude that there is a statistically significant difference between the 8 models ($p < 2.2e^{-16}$). To then pinpoint where the difference precisely lies, pairwise comparisons using Student's t-test show that the difference between ILP (using semantics) and all of the other learning models is statistically significant at $p < 0.05$. Additionally, ILP with semantics is significantly better than ILP without it ($p < 0.05$). For Decision Tree, Naive Bayes and RIPPER, the improvement in using semantics is not statistically significant.

Model	Semantics	No Semantics
ILP	**59.43**	59.22
Decision Tree	53.84	53.69
RIPPER	51.1	51.36
Naive Bayes	49.69	51.62
Majority Class	22.01	22.01

Table 2: Classification Performance: Set B (F-Score)

In Table 2, we list the results on Set B. Once again, our ILP model outperforms the other three learning models. Naive Bayes is much more competitive when using all the examples compared to using only examples with semantic data. In the case of the attribute-value machine learning models, the use of semantic data seems to marginally hurt the performance of the classifiers. However, this is in contrast to the relational ILP model which always performs better when using semantics. This result suggests that the use of semantic data with loss of information may not be helpful, and in fact, it may actually hurt performance. Based on ANOVA, the differences in these 8 models is statistically significant with $p < 6.95e^{-12}$. A pairwise t-test between ILP (using semantics) and each of the other attribute-value learning models shows that our results are statistically significant at $p < 0.05$.

In Table 3, we report the performance of the two ILP models on each relation.[3] In general, the models perform better on relations that have the most examples.

The evaluation of work in discourse parsing is hindered by the lack of a standard corpus or task. Hence, our results cannot be directly compared to (Marcu, 2000; Sporleder and Lascarides, 2005; Wellner et. al., 2006), but neither can those works be compared among themselves, because of differences in underlying corpora, the type and number of relations used, and various assumptions. However, we can still draw some general comparisons. Our ILP-based models provide as much or significantly

[3] Due to space limitations, only relations with > 10 examples are shown.

relation	Semantics	No Semantics
preparation:act	74.86	72.05
general:specific	31.74	28.24
joint	55.23	52
act:goal	86.12	83.85
criterion:act	77.37	75.32
goal:act	73.43	68.9
step1:step2	28.75	35.29
co-temp1:co-temp2	48.84	37.84
disjunction	83.33	80.81
act:criterion	54.29	54.79
contrast1:contrast2	22.22	5.0
act:preparation	65.31	70.59
act:reason	0	10.26
cause:effect	19.05	10.53
comparison	22.22	10.53

Table 3: Classification Performance (F-Score) by Relation: ILP on Set A

more improvement over a majority-class baseline when compared to these other works. This is the case even though our work is based on less training data, relatively more relations, relations both between just two EDUs and those involving larger text segments, and we make no assumptions about the order of the relations. Our results are comparable to (Marcu, 2000), which reports an accuracy of about 61% for his classifier. His majority class baseline performs at about 50% accuracy. (Wellner et. al., 2006) reports an accuracy of up to 81%, with a majority class baseline performance of 45.7%. However, our task is more challenging than (Wellner et. al., 2006). They use only 11 relations compared to the 26 we use. They also assume the order of the relation in the examples (i.e. examples for *goal:act* would be treated as examples for *act:goal* by reversing the order of the arguments) whereas we do not make such assumptions. In addition, their training data is almost twice as large as ours, based on relation instances. (Sporleder and Lascarides, 2005) also makes the same assumption on the ordering of the relations as (Wellner et. al., 2006). They report an accuracy of 57.75%. Their work, though, was based on only 5 relations. Importantly, neither (Wellner et. al., 2006; Sporleder and Lascarides, 2005) model examples with complex text segments with more than one EDU.

5.2 How interesting are the rules?

Given that our ILP models learn first-order logic rules, we can make some qualitative analysis of the rules learned, such as those below, learnt by the ILP model that uses semantics:

```
(2a) relation(A,B,'act:goal') :-
     firstWordPOS(A,'VBG'),
     verbclass(A,D,'use-1'),
     firstWordPOS(B,'VB').
     [pos cover = 23 neg cover = 1]

(2b) relation(A,B,'preparation:act') :-
     discourse_cue(B,front,and),
     cause(A,frame(C),D,E),
     theme(B,frame(F),G), theme(A,frame(C),G).
     [pos cover = 12 neg cover = 0]

(2c) relation(A,B,'preparation:act') :-
     discourse_cue(B,front,then),
     parent_segment(A,C), parent_segment(A,D),
     internal_relation(C,D,'preparation:act').
     [pos cover = 17 neg cover = 0]
```

(2a) is learned using examples such as *relation(*$s5e1$,$s5e2$*,'act:goal')* from Example (1). (2b) uses relational semantic information. This rule can be read as follows:

> **IF** segment A contains a cause and a *theme*, the same object that is the *theme* in A is also the *theme* in segment B, and B contains the discourse cue *and* at the front **THEN** the relation between A and B is *preparation:act*.

(2c) is a rule that makes use of the structural information about complex text segments. When using Set A, more than about 60% of the rules induced include at least one semantic predicate in its body. They occur more frequently in rules for relations like *preparation:act* while less in rules for *general:specific* and *act:goal*.

5.3 Discourse Parsing Results

In order to test our discourse parser, we used 151 documents for training and 25 for testing. We evaluated the performance of our parser on both the discourse parse trees it builds at the sentence level and at the document level. The test set contained

		Sentence Level			Document Level		
model	Semantics	span	nuclearity	relation	span	nuclearity	relation
SR-ILP	yes	92.91	71.83	**63.06**	**70.35**	**49.47**	**35.44**
SR-ILP	no	91.98	69.59	58.58	68.95	48.16	33.33
Baseline	-	**93.66**	**74.44**	34.32	70.26	47.98	22.46

Table 4: Parsing Performance (F-Score): (Baseline = right-branching majority)

341 sentences out of which 180 sentences were segmented into more than one EDU. We ran experiments using our two ILP models for the relation identifier, namely ILP with semantics and without semantics. Our ILP based discourse parsing models are named SR-ILP. We compare the performance of our models against a right branching majority class baseline. We used the sign-test to determine statistical significance of the results. Using the automatic evaluation methodology in (Marcu, 2000), precision, recall and F-Score measures are computed for determining the hierarchical spans, nucleus-satellite assignments and rhetorical relations. The performance on labeling relations is the most important measure since the results on nuclearity and hierarchical spans are by-products of the decisions made to attach segments based on relations.

On labeling relations, the parser that uses all the features (including compositional semantics) for determining relations performs the best with an F-Score of 63.06%. The difference of about 4.5% (between ILP with semantics and without semantics) in F-Score is statistically significant at $p = 0.006$. Our best model, SR-ILP (using semantics) beats the baseline by about 28% in F-Score. Since the task at the document level is much more challenging than building the discourse structure at the sentence level, we were not surprised to see a considerable drop in performance. For our best model, the performance on labeling relations drops to 35.44%. Clearly, the mistakes made when attaching segments at lower levels have quite an adverse effect on the overall performance. A less greedy approach to parsing discourse structure is warranted.

While we would have hoped for a better performance than 35.44%, to start with, (Forbes et. al., 2001), (Polanyi et. al., 2004), and (Cristea, 2000) do not report the performance of their discourse parsers at all. (Marcu, 2000) reports precision and recall of up to 63.2% and 59.8% on labeling relations using manually segmented EDUs on three WSJ articles. (Baldridge and Lascarides, 2005) reports 43.2% F-Score on parsing 10 dialogues using a probabilistic head-driven parsing model.

6 Conclusions

In conclusion, we have presented a relational approach for classifying informational relations and a modified shift-reduce parsing algorithm for building discourse parse trees based on informational relations. To our knowledge, this is the first attempt at using a relational learning model for the task of relation classification, or even discourse parsing in general. Our approach is linguistically motivated. Using ILP, we are able to account for rich compositional semantic data of the EDUs based on VerbNet as well as the structural relational properties of the text segments. This is not possible using attribute-value based models like Decision Trees and RIPPER and definitely not using probabilistic models like Naive Bayes. Our experiments have shown that semantics can be useful in classifying informational relations. For parsing, our modified shift-reduce algorithm using the ILP relation classifier outperforms a right-branching baseline model significantly. Using semantics for parsing also yields a statistically significant improvement. Our approach is also domain independent as the underlying model and data are not domain specific.

Acknowledgments

This work is supported by the National Science Foundation (IIS-0133123 and ALT-0536968) and the Office of Naval Research (N000140010640).

References

Asher, N., and Lascarides, A.: Logics of Conversation. Cambridge University Press, 2003.

Baldridge, J. and Lascarides, A.: Probabilistic Head-Driven Parsing for Discourse Structure In Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNNL), Ann Arbor, 2005.

Brill, E.: Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. Computational Linguistics, 21(4):543565, 1995.

Buitelaar, P.: CoreLex: Systematic Polysemy and Underspecification. Ph.D. Thesis, Brandies University, 1998.

Carlson, L. D. M. and Okurowski., M. E.: Building a discourse-tagged corpus in the framework of rhetorical structure theory. In Current Directions in Discourse and Dialogue pages 85–112, 2003.

Cristea, D.: An Incremental Discourse Parser Architecture. In D. Christodoulakis (Ed.) Proceedings of the Second International Conference - Natural Language Processing - Patras, Greece, June 2000.

Forbes, K., Miltsakaki, E., R. P. A. S. A. J. and Webber., B.: D-ltag system - discourse parsing with a lexicalized tree adjoining grammar. Information Stucture, Discourse Structure and Discourse Semantics, ESSLLI 2001.

Grosz, B. J. and Sidner, C. L.: Attention, intention and the structure of discourse. Computational Linguistics 12:175–204, 1988.

Hobbs, J. R.: On the coherence and structure of discourse. In Polyani, Livia editor, The Structure of Discourse, 1985.

Kipper, K., H. T. D. and Palmer., M.: Class-based construction of a verb lexicon. AAAI-2000, Proceedings of the Seventeenth National Conference on Artificial Intelligence, 2000.

Mann, W. and Thompson, S.: Rhetorical structure theory: Toward a functional theory of text organization. Text, 8(3):243–281, 1988.

Marcu, D.: Instructions for Manually Annotating the Discourse Structures of Texts. Technical Report, University of Southern California, 1999.

Marcu, D.: The theory and practice of discourse parsing and summarization. Cambridge, Massachusetts, London, England, MIT Press, 2000.

Moser, M. G., Moore, J. D., and Glendening, E.: Instructions for Coding Explanations: Identifying Segments, Relations and Minimal Units. University of Pittsburgh, Department of Computer Science, 1996.

Muggleton, S. H.: Inverse entailment and progol. In New Generation Computing Journal 13:245–286, 1995.

Polanyi, L., Culy, C., van den Berg, M. H. and Thione, G. L.: A Rule Based Approach to Discourse Parsing. Proceedings of the 5th SIGdial Workshop in Discourse And Dialogue. Cambridge, MA USA pp. 108-117., May 1, 2004.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B.: The Penn Discourse Treebank 2.0. LREC, 2008.

Rosé, C. P.: A Syntactic Framework for Semantic Interpretation, Proceedings of the ESSLLI Workshop on Linguistic Theory and Grammar Implementation, 2000.

Sporleder, C. and Lascarides., A.: Exploiting linguistic cues to classify rhetorical relations. Recent Advances in Natural Language Processing, 2005.

Soricut, R. and Marcu., D.: Sentence level discourse parsing using syntactic and lexical information. Proceedings of the Human Language Technology and North American Assiciation for Computational Linguistics Conference, 2003.

Subba, R., Di Eugenio, B., E. T.: Building lexical resources for princpar, a large coverage parser that generates principled semantic representations. LREC, 2006.

Subba, R.: Discourse Parsing: A Relational Learning Approach Ph.D. Thesis, University of Illinois Chicago, December 2008.

Webber, B.: DLTAG: Extending Lexicalized TAG to Discourse. Cognitive Science 28:751-779, 2004.

Wellner, B., Pustejovsky, J., C. H. R. S. and Rumshisky., A.: Classification of discourse coherence relations: An exploratory study using multiple knowledge sources. In Proceedings of the 7th SIGDIAL Workshop on Discourse and Dialogue, 2006.

Williams, S. and Reiter, E.: A corpus analysis of discourse relations for natural language generation. Proceedings of Corpus Linguistics, pages 899–908, 2003.

Wolf, F. and Gibson, E.: Representing discourse coherence: A corpus-based analysis. Computational Linguistics 31(2):249–287, 2005.