Constrained decoding for text-level discourse parsing

Philippe Muller, Stergos Afantenos, Pascal Denis, Nicholas Asher

Constrained decoding for text-level discourse parsing

Philippe Muller[1] *Stergos Afantenos*[1] *Pascal Denis*[2] *Nicholas Asher*[3]
(1) IRIT, Université de Toulouse, France
(2) Mostrare, INRIA, France
(3) IRIT, CNRS, France
{stergos.afantenos,muller,asher}@irit.fr, pascal.denis@inria.fr

Abstract

This paper presents a novel approach to document-based discourse analysis by performing a global A* search over the space of possible structures while optimizing a global criterion over the set of potential coherence relations. Existing approaches to discourse analysis have so far relied on greedy search strategies or restricted themselves to sentence-level discourse parsing. Another advantage of our approach, over other global alternatives (like Maximum Spanning Tree decoding algorithms), is its flexibility in being able to integrate constraints (including linguistically motivated ones like the Right Frontier Constraint). Finally, our paper provides the first discourse parsing system for French; our evaluation is carried out on the Annodis corpus. While using a lot less training data than earlier approaches than previous work on English, our system manages to achieve state-of-the-art results, with F1-scores of 66.2 and 46.8 when compared to unlabeled and labeled reference structures.

Keywords: Discourse Structure, Discourse Parsing, Dependency Structures, Constrained Decoding, A*.

1 Introduction

Discourse analysis involves the identification of coherence relations between discourse units, which can be either elementary ones—typically clauses or sentences—or complex ones, spanning larger chunks of text. These larger units play similar roles to elementary ones. Coherence relations categorize discourse units in terms of their argumentative, thematic or causal links to other units. Together these relations and the units they relate form the global structure of a discourse.

This structure is important as it reflects the thematic organization at various levels of granularity, and constrains semantic interpretations, for instance with respect to anaphora resolution or temporal interpretation. Discourse processing is thus a crucial part of natural language understanding, and it has potentially many applications—opinion detection and classification, question answering, information extraction, recognizing textual entailment, evaluating text coherence, or knowledge extraction to name a few (Stede, 2011; Verberne et al., 2007; Somasundaran et al., 2009; Lin et al., 2011; Gerber et al., 2010)

Producing a discourse structure automatically is a complex task, however. Coherence involves syntactic and lexical factors, and relies heavily on the semantic interpretation of its parts. Most existing work tends to focus on restricted aspects of the full problem.

The task of building a discourse structure involves three subtasks: (1) identifying discourse units (DUs), (2) "attaching" DUs to one another, and (3) labeling their link with a coherence relation. Of these, the first one is usually considered easiest (see for instance Hernault et al., 2010), the second is arguably the hardest; and as a consequence, researchers have focussed attention on the third one, labelling discourse relations (Lin et al., 2009; Feng and Hirst, 2012; Sporleder and Lascarides, 2008; Wellner et al., 2006; Louis et al., 2010). Research on discourse structure also divides into two orthogonal categories: some researchers limit themselves to intra-sentential discourse structure (Wellner et al., 2006; Sagae, 2009; Joty et al., 2012); others tackle the problem of identifying the full discourse structure of a text (Hernault et al., 2010; Subba and Di Eugenio, 2009). The latter rely on "local" models to predict potential coherence relations, assuming independence between the decisions, and build the structure guided by greedy heuristics. The exception is (Baldridge and Lascarides, 2005), who use a generative model, in the case of specific task-oriented dialogues.

In this paper, we propose a more general approach to discourse structure prediction at the document level: (i) it performs a global search over the space of possible structures and optimizes a global criterion over the set of potential coherence relations; (ii) it can also take into account linguistically motivated constraints on the predicted structure. Specifically, our approach relies on the A* search algorithm, which is particularly well suited in allowing to capture constraints such as the so-called Right Frontier Constraint (or RFC), various versions of which have been a staple of theoretical linguistic approaches to discourse (Polanyi, 1988).

Another contribution of our paper is to provide a simple formalism that captures many of the commonalities across particular representation models for full discourse structure by considering a more general graph-based model, which can nevertheless integrate framework specific constraints. Previous work relies on several different discourse representation theories and corpora—e.g., the Penn Discourse Tree Bank (PDTB) of (Prasad et al., 2008), which assumes a light-weight linear structure; Rhetorical Structure Theory (RST) and the RST treebank (Carlson et al., 2003), which assumes a constituent tree structure; Segmented Discourse Representation Theory (SDRT) from which derive the Discor and Annodis corpora

(Baldridge et al., 2007; Afantenos et al., 2012), with directed acyclic graphs; or GraphBank (Wolf and Gibson, 2006), with little if any constraints on a graph-based structure.

Incidentally, we also deliver the first discourse parsing system for French. Our evaluation is performed on the ANNODIS corpus (Afantenos et al., 2012).

The paper is structured as follows. Section 2 positions in more detail our work with respect to the existing literature on discourse parsing. Section 3 describes our approach to discourse structure "decoding". Section 4 introduces the data we use from the Annodis corpus, and sections 5-6 present our experiment design. Finally section 7 reports our results and an analysis, especially with respect to comparable work.

2 Related work

Apart from a few exceptions, research on automatic discourse analysis has focused on specific aspects of the general problem. Most work concerns the task of discourse relation labeling between pairs of DUs. Examples of this line of work are: Marcu and Echihabi (2002), Sporleder and Lascarides (2005) and Lin et al. (2009). This setting makes an unwarranted assumption, as it assumes one decision concerning labeling is independent from another. Alternatively, researchers have considered the task of predicting full discourse structures, but only at the sentence level. An example of sentence-level discourse parsing is Soricut and Marcu (2003), which makes use of dynamic programming along with a standard bottom-up chart parsing. In this case, probabilities for each sentence level discourse tree are calculated as the product of the probabilities for the structure and the relation. More recently, Sagae (2009) has developed a shift-reduce algorithm for intra-sentential discourse analysis. Their stack contains the current discourse subtree and it consumes a sequence of EDUs. Like Soricut and Marcu (2003), Sagae (2009) use the RST Discourse Treebank. RST trees are "lexicalized" through head percolation using the so-called nucleus/satellite distinction. The shift operation removes the next EDU from the sequence and pushes a subtree containing only that EDU onto the stack. The reduce operation is either unary or binary: unary reduce just pops the stack and pushes a subtree with the popped item as the only child and it's lexicalized head as its mother, while the left and right binary reduce operations pop two nodes, attach them and push them back—left and right being used to judge nuclearity.

There are two main reasons why the full task of discourse parsing has eluded NLP researchers. The first is that annotating discourse structures is a very expensive procedure. There are but modest amounts of data that have been annotated, and structured prediction views each document as a single instance. Consequently, training is not very reliable. The second reason is that the two largest discourse-annotated corpora (the PDTB and the RST corpus) enforce strong constraints on the structure (namely attachment to adjacent DUs) and are thus naturally biased toward local approaches where only attachment to the left or right DU should be considered, ignoring the interdependence of local decisions. This problem can be tackled more easily at the sentence-level, where structures are simpler with only a few discourse units. Sentences can be considered independently of one another, and provide more training instances and more reliable predictions.

Among the few attempts to build document-level discourse parsers are Subba and Di Eugenio (2009) and duVerle and Prendinger (2009). Like Sagae (2009), Subba and Di Eugenio (2009) use a transition-based approach. As in the intra-sentential work cited above, the

shift operation places the next segment on top of the stack, but there is only a *binary* reduce operation which may result in the triggering of more reduce operations. In case no reduce operation is triggered then a shift is automatically performed. Consequently, only the reduce operations need to be learned. In case that the input string is empty but the stack is not, a reduce with the relation LIST is (continuously) performed. Subba and Di Eugenio (2009) use rich linguistic features and Inductive Logic Programming for training. All results are reported on an in-house corpus, and they barely surpass the baseline that consists in always attaching to the last DU.

duVerle and Prendinger (2009), and its sequel Hernault et al. (2010) both rely on locally greedy methods, and in line with all previous works, treat attachment prediction and relation label prediction as independent problems. Specifically, they start by computing probabilities of attachments for adjacent pairs of EDUs and greedily select the highest scoring. A second classifier, applied in cascade, determines the relation for the two DUs. The pair is replaced by the created "span" and the procedure continues in a bottom up way. The recent work of Feng and Hirst (2012) extend this approach by additional feature engineering but is restricted to sentence-level parsing. These three papers all use the RST-DT.

Joty et al. (2012) deserve special mention because they consider inter-dependence of local decisions, but nonetheless limit themselves on the level of intra-sentential parsing. As a first step, they compute the *joint* probabilities for structure plus relation for all possible combinations of structures and relations within a single sentence. For the computation of those joint probabilities they use a Dynamic CRF. Once all the possible joint probabilities have been computed, they perform a classic CKY chart parsing using dynamic programming. They use features representing text organization, dominance sets, contextual information and hierarchical dependencies. This is a very interesting approach but it does not easily scale up for the whole text, at least using a CRF-like approach. This is where we distinguish ourselves by adopting a local model and then a constraint-based decoding mechanism for identifying the optimal global structure.

Another relevant paper (Baldridge and Lascarides, 2005) presents a comparable problem, namely predicting the rhetorical structure of dialogues, from the Verbmobil corpus. Dialogues are considered as documents with relations between utterances, and the authors train a PCFG to produce tree representations of the dialogue structure. It is unclear how this approach, which works on very specific task-oriented dialogues, would perform in a more general framework, especially since they require some semantic features that were annotated manually.

3 Discourse decoding under constraints

In order to recover the rhetorical structure of a document, we take as our starting point two locally-trained classifiers predicting the attachment of discourse units and the labelling of their relations, much in the same way as (Hernault et al., 2010; Subba and Di Eugenio, 2009). Our models are also "local" in the sense that the training criterion that they optimize is still local and the features they use are defined over pairs of DUs. But we differ from previous approaches in the way we use the outputs of the local classifiers to predict the overall discourse structure, as well as in the use of constraints (such as the Right Frontier Constraint) during this "decoding" phase. Yet another difference is that we predict attachment decisions and relation labeling decisions in a joint fashion, rather than in pipeline as it is done in previous work.

Before getting into the details of the decoding, let's first consider the type of structure we intend to produce. An important challenge with discourse analysis is the lack of consensus among discourse theorists as to the relevant type of representations that one should use to encode discourse structure. These theoretical differences are directly reflected in the various existing discourse annotation corpora, as most of them are based on one particular theoretical framework. As a result, the different existing systems being trained on a specific resource are framework-specific. One of our goals in this paper is to abstract away from these differences and provide a more generic approach to the problem of discourse parsing.

Consider RST-DTs. RST representations are similar to constituent-based syntactic trees, since relational structures are recursively built bottom-up from elementary discourse units to form complex discourse units and adjacency is enforced at each level. Users of RST also often assume the so-called "nuclearity principle", by which complex RST segments have a distinguished EDU as a sort of head, a procedure similar to head percolation in syntax, when converting to a constituent-based to a dependency based representation. Other frameworks like SDRT or GraphBank assume more general structures (respectively directed acyclic graphs and graphs). One uniform way to capture commonalities between these different types of representation is to convert them into a dependency graph between EDUs. Many of the differences between frameworks can be encoded with different, additional structural constraints. To capture RST up to complex segments, one translates an RST tree recursively taking the nuclearity principle into account. SDRT also has complex segments, and we address how they can be dealt with in section 4. In SDRT, discourse interpretation is supposed to be an incremental process that respects a "right frontier constraint" (RFC) (Polanyi, 1988): discourse units are supposed to be processed one by one, and the current unit can only be attached to a node on the "right frontier" formed by the last introduced node and nodes "above" it (assuming a hierarchical structure). These theories distinguish between relations that are "coordinating" (additive relations, also called multinuclear in RST) or "subordinating" (expanding relations, or nuclear-satellite in RST). In that case subordinating relations add a level to the discourse hierarchy while coordinating stay at a given level of granularity.

Dependency structures have become very prominent for *syntactic* parsing, and a number of approaches have been applied successfully to the problem. So, a natural question is whether techniques developed therein could be directly used for discourse analysis. As noted, there have been some initial attempts at adapting shift-reduce parsers to discourse. An alternative to transition-based parsing is the graph-based parsing proposed e.g. by McDonald et al. (2005). This approach is particularly appealing since it builds upon an exact search procedure for finding the best possible dependency tree: it is the Maximal Spanning Tree (or MST) on the fully connected graph defined on the sentence words. One could in principle use this approach for discourse parsing, but as with transition-based parsing, there is no obvious way to easily integrate global constraints as the RFC.

Our solution is to express the problem of discourse parsing as a state-space search with different state-space definitions and constraints, and apply a general A* exploration strategy. A* search is shortest-path search through the state of possible results (dependency graphs in our case), which orders the search considering an estimated cost of a partial solution as the sum of the cost of the part of the solution already built and the estimated cost of the remaining part to be built. Transitions between states should be here the choice of an edge between two DUs, to be added to the desired solution. Since transition costs must

be additive, the cost of an edge will be -log its probability, as given by the models for attachment and relations. The general form of such a search is shown as algorithm 1. The estimation, or "heuristics", guarantees an optimal solution under certain conditions, the most common being that the heuristics is "admissible", i.e. it always underestimates the cost of the remaining exploration.

Algorithm 1 General A* decoding. S_0 is an initial state, problem dependent. Functions g and h are the cost of 'current' so far, and the estimated cost. An example of state generation is shown in algorithm 2

```
procedure astarSearch(S_0)
    queue ← {S_0}
    while not(queue.isEmpty()) do
        current ← removeBest(queue)                    ▷ best according to g+h(current)
        if isSolution(current) then return current
        else
            newStates ← generate(current)              ▷ well formed wrt desired constraints
            queue = queue ∪ newStates
        end if
    end while
end procedure
```

In contrast to the greedy approaches of (Hernault et al., 2010; Subba and Di Eugenio, 2009), we have more control on the solution yielded by the procedure. With an admissible heuristics, A* guarantees an optimal solution with respect to the cost function on state transitions (here, the probability of a given relation between two discourse units). Limited search can further be implemented as a special case if the state-space proves to be combinatorial, by restricting A* with a beam (a pending queue of fixed size, but then losing completeness). A* has also been used in syntactic parsing because of these advantages (Pauls and Klein, 2009). Another advantage offered by A* search and used in the previous paper, lies in its ordering of hypotheses, that easily yields the n best solutions by continuing the exploration of the state space. This is useful for instance to apply reordering or ranking techniques.

The most delicate aspect of using A* is in the heuristics chosen to guide the search. We will discuss possible heuristics in section 6, as they are rather orthogonal to the current discussion.

The state-space exploration works as an incremental building of a solution and must specify mainly: (1) a starting state for building a solution and (2) allowed states from a given state. For instance an MST approach could be implemented (inefficiently, though) with a starting state consisting of just one discourse unit, or as a fake node related to the others depending on the probability of a segment to be the head of the discourse (as is done in syntactic parsing). Following states could then only add a relation between a new node and one and only one of the already chosen nodes, until all nodes are attached.

To implement the RFC, we only need to restrict the previous procedure so that new nodes can only be attached to the set of accessible nodes assuming the RFC, see for details algorithm 2.

In case one wants to stay within the RST framework, the starting state would be empty, and new states should be built by adding a relation between two adjacent active units. Active units are all elementary units at the beginning, each EDU being replaced by a complex one

as they are attached during the decoding procedure.

Algorithm 2 Example state generation for building a tree incrementally under the RFC constraint, taking the first segment as the root. For simplicity, (1) relations are ignored since the best for each edge are incorporated in the cost evaluations, and (2) no difference is made between additive or expanding relation for updating the right frontier RF. The treatment below correspond to expanding relations only. For additive relations, the new attached unit replaces the attachment point in the RF, so the RF update should be new.RF = new.RF[:k-1] + [next].

Let I = $[u_1, ..., u_n]$ the list of elementary discourse units, in text order.
And let a general state S = $\langle V, E, RF \rangle$ where V is the list of DU to be attached, E is a set of edges making up a tree, and RF is a list of accessible nodes according to the right frontier constraint. We will note the state parts as S.V, S.E and S.RF.
Initially, S = $\langle I[2:], \emptyset, [u_1] \rangle$, as we take the first segment as the root. The following will generate a tree structure respecting the RFC, approximating a SDRT DAG:

```
procedure generate(S)
    result = ∅
    next = head(S.V)
    for k in 1 to len(S.RF) do        ▷ This loop generate hypotheses attaching 'next' to every u ∈ RF.
        new = newState()
        new.V = pop(copy(S.V))                ▷ Next is removed from the nodes to attach.
        new.E = E ∪ (next,S.RF[k])            ▷ The attachment is added to the current structure.
        new.RF = new.RF[:k] + [next]          ▷ Next is appended to the RF after its parent, everything
                                                  else below the parent is thrown out.
        result.add(new)
    end for
    return result
end procedure
```

4 The corpus used

A number of different corpora have been annotated with discourse structures. These differ in the discourse formalism they are grounded in, and from our perspective in the type of constraint they impose on the attachment of DUs. The one with the heaviest constraints on discourse attachment, and consequently the simplest structures, is PDTB (Prasad et al., 2008), as most attachment are between *adjacent EDUs*, creating no further complex structures. To be fair, the main focus of PDTB is not discourse structure *per se*, but instead the study of explicit and implicit discourse relations—as signaled by discourse markers or absence thereof. Based on *Rhetorical Structure Theory* (RST) (Mann and Thompson, 1987; Marcu, 2000), the RST Discourse Treebank (RST-DT) does have recursive structures, but it imposes heavy constraints by still enforcing adjacency (Marcu, 2000). More specifically, in RST an EDU can be attached either to its adjacent EDUs (forming what is called a *span*) or to any other adjacent span, recursively thus creating a tree.[1] Less constrained structures are found in two other approaches, first from Wolf and Gibson (2006) (the GraphBank corpus) which creates graph structures with apparently no constraints whatsoever, second from SDRT (Asher and Lascarides, 2003) which creates directed acyclic graphs imposing only the RFC. Two different annotations campaigns have used the SDRT framework: DISCOR (Baldridge et al., 2007) for English and ANNODIS (Afantenos et al., 2012) for French.

[1] It can be the case that more than two DUs (*always adjacent*) could be attached together for relations such as LIST.

We have used the ANNODIS corpus: it is a collection of French discourse annotated newspaper and Wikipedia articles; specifically, we used the so-called "expert" annotations from the sub-corpus that deals with rhetorical relations. The chief reason for choosing a corpus based on SDRT is that it provides a compromise between simplistic approaches to discourse (attachment on adjacent DUs) and completely unrestricted approaches (such as the GraphBank corpus). SDRT manages to capture fine grained discourse phenomena, such as for example long distance attachments and pop-ups, while at the same time imposing a few constraints through the distinction between hierarchical and additive relations, such as the right frontier constraint.

The relation set used in the ANNODIS annotation campaign is a strict subset of the set of relations described in Asher and Lascarides (2003) (for example, there are no meta-relations); their semantics was also simplified in order to be accessible to naive subjects. Relation distribution is shown in table 1.

relation name	#	%	relation name	#	%
alternation	18	0.5	explanation	130	3.9
attribution	75	2.2	flashback	27	0.8
background	155	4.6	frame	211	6.3
comment	78	2.3	goal	95	2.8
continuation	681	20.3	narration	349	10.4
contrast	144	4.3	parralel	59	1.8
Eelab	527	15.7	result	163	4.9
elaboration	625	18.6	temploc	18	0.5
total # relations		3355	total # EDUs		3188
total # CDUs		1395	total # texts		86

Table 1: Corpus statistics from the ANNODIS corpus

In order to be able to apply techniques from syntactic dependency parsing, we transformed SDRT Annodis annotations into dependency graphs by replacing complex discourse units with their *recursive heads*. Annotations indeed mix EDU (elementary DU) and sets of EDUs, which are comparable to large spans in RST, with less constraints on their members, and this procedure is then another kind of head percolation. The *head* of a CDU is the highest DU in its subgraph (in terms of hierarchical/subordinate relations) and leftmost or older DU in the discourse with respect to additive/coordinate relations if there is more than one. In case that the DU is a complex one, the procedure is recursively applied until an EDU is reached, in which case that is the recursive head of the CDU. This transformation is graphically depicted in figure 1 which also contains the corresponding text and segmentation in the original French language.

5 Local models

Our discourse parsing is based on two locally-trained classifiers, one that predicts the attachment site of each DU, the other that predicts a discourse relation for attached pairs of DUs. In both cases, we trained probabilistic classifiers, using two different types of model: Naive Bayes (NB) and logistic regression (aka maximum entropy, or MaxEnt for short).[2]

[2]Pamameter estimation for the latter was performed using (Daumé III, 2004), `http://www.cs.utah.edu/~hal/megam/`. We also used utilities provided by the Orange library of (Curk et al., 2005).

[Principes de la sélection naturelle.]_1 [La théorie de la sélection naturelle [telle qu'elle a été initialement décrite par Charles Darwin,]_2 repose sur trois principes:]_3 [1. le principe de variation]_4 [2. le principe d'adaptation]_5 [3. le principe d'hérédité]_6

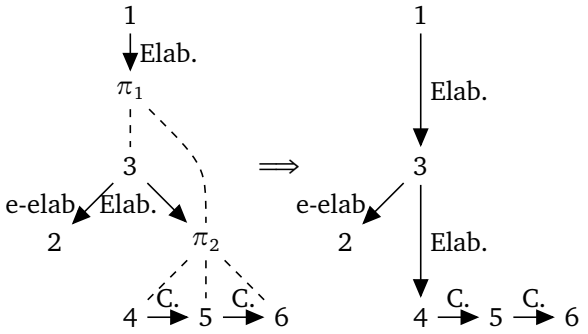

Figure 1: **An example of discourse annotation**. The nodes correspond to discourse units; the EDUs are represented by their numbering; the CDUs start with π. Dotted edges represent inclusion to a CDU while edges with arrows represent rhetorical relations. Elab. = Elaboration, e-elab = Entity Elaboration, C. = Continuation. The second graph is the result of the transformation.

The use of probabilistic models is guided by the way we combine the two models during decoding, and will be explained in Section 6.

We used two different, partially overlapping, feature sets for attachment and labeling. Overlapping features reflect an inclusion in the same sentence or paragraph, an EDU being the first of the paragraph, the number of tokens in an EDU, the number of intervening EDUs between source and target EDUs, whether the source is embedded in the target and conversely. Features specific to attachment include the presence of a discourse marker, whether the target is embedded in an EDU other than the source and a boolean feature triggered by a set of syntactic rules determining whether source (or target) is an apposition or relative clause embedded in its main clause. Features specific to labeling include: the presence of a verb, boolean features indicating which discourse relations are triggered from all discourse markers in the EDU, the syntactic category of the head token, the presence of a negation, tense agreement between head verbs of both source and target (the last three make use of syntactic dependency parses[3]) and features inspired from coreference resolution (based on pronouns and NPs).

Evaluation for the two tasks, based on 10-fold cross validation on the document level, is shown in table 2. For the relation labeling task, we use the whole set of 18 relations annotated in the Annodis corpus, but since this is a relatively small corpus, we also considered a smaller sets of relations. Following other hierarchies of relations (RST and PDTB), we grouped SDRT-inspired relations into four main groups: "structural" (parallel, contrast, alternation, conditional); "sequence" (result, narration, continuation); "expansion" (frame, elaboration, e-elaboration, attribution, comment, explanation); and "temporal" (temploc, goal, flashback, background). This corresponds roughly to the PDTB upper-level 4-way distinction, namely temporal, causal ("contingency"), comparison and expansion, with comparison being almost the same as structural without the logical relations, but the sets of fine-grain relations are somewhat different. Our 4-way coarse grain classification is also

[3]We have used Malt as a syntactic parser, trained on the french treebank `http://alpage.inria.fr/statgram/frdep/fr_stat_dep_malt.html`

more evenly distributed between relations (and instances, as it turns out).

For both tasks, we perform our experiments using a set that contains all possible pairs of EDUs, and a set that considers only pairs of EDUs whose distance is between 1 and 5 (noted "w5" for window of 5 in the table). This window was decided upon using a small development test, whose analysis revealed that around 92% of the attachment decisions fell within a window of 5 DUs. The class imbalance inherent to the attachment problem was thus reduced: the ratio of positive instances (i.e., attachments) went up roughly from 2% to 20%.

	MaxEnt	NB	Majority
w5 (18 relations)	**44.8**	34.7	19.1
full (18 relations)	43.3	32.9	19.7
w5 (4 relations)	**65.5**	62.1	51.2
full (4 relations)	63.6	60.1	50.1

	MaxEnt	NB
w5	**67.4**	61.1
full	63.5	51.3

Table 2: Relation classification accuracy (left) and F1 score for positive attachment (right), in %. For both classifications tasks the difference between Maxent and Naive Bayes is significant at $p<0.01$, using McNemar's test. The upper limit recall for the latter task in w5 configuration is 92%.

These results show that Maxent is the best model in isolation, for both tasks (it is better in both precision and recall). As expected, we also notice that the resampling increases performance in both cases; although it isn't reported here, it does however produce a small decrease in recall for the attachment task. On this task, Maxent appears to be more robust to class imbalance than NB, as shown by the relative differences between attachment F1-scores with and without resampling.

6 Parsing experiments

We divided the ANNODIS corpus in two parts: a main part and a small development set on which we had a look on the impact of some features, and most importantly on the distribution of distances between discourse units actually attached. As explained in Section 5, this leads to different sampling strategies for training the local classifiers. This will also impact decoding in pre-pruning the hypothesis space.[4]

Our various experiments are based on different combinations of classifier models (as detailed in the previous section) and decoding strategies. For attachment, we consider as instances either every pair of DUs in the same text or every pair in a distance equal to five or less. For labeling, the training is made only on attached discourse units in the training set and predictions are made on every pair tested for attachment. Features for each training procedures were detailed in section 5. As decoders, we tested a few baselines as well as MST and A*, all detailed further in this section. The last two algorithms take as input G, the complete graph over discourse units, where each edge (u, v) is labelled with the relation R having the best probability according to the relation labeling model, and the cost or weight

[4]Note that DUs are always ordered based on their left boundary. This will be important for A* decoding, which needs an ordered set of DUs in order to respect the right-frontier constraint. Practically, it means that embedded segments are attached only once their containing segment has been processed. An embedded segment is considered to be at distance one to its container.

of an edge is given by:

$$cost((u,v)) = -\log(Pr(attach(u,v) = True) \times max_R Pr(R|attach(u,v) = True))$$

This way of computing each arc cost means that we are in effect taking attachment and labeling decisions jointly, and not in a cascade as is done in the baselines.

Baselines We use two baseline decoders. The first one always selects the previous unit for attachment to the current one (noted "last"). We have also implemented a locally greedy approach similar to that of (Hernault et al., 2010). DUs are ordered based on their left boundary (embedded segments are considered to be at distance 1 from their containers). Then for all pair of adjacent units[5] (e_i, e_j) we greedily select the one that has the highest attachment probability. We remove e_i from the ordered list and continue the process until there is only one unit left in the list.

Using the Chu-Liu Edmonds algorithm The structures that result from the replacement of CDUs with their recursive heads in Annodis annotations are directed acyclic graphs, with few edges reaching a given node; they are thus very close to non-projective trees with directed arcs. Naturally then, we can apply a Maximum Spanning Tree (MST) approach, as applied by McDonald et al. (2005) in the context of syntactic dependency parsing for directed non-projective dependency trees. MST can consider an almost complete graph with a "root" as the only node with no incoming edges. In our case this node will be the first (leftmost) EDU. Using either NB or MaxEnt, we calculate the probabilities of attachment between each pair of EDUs, except for the first (root) EDU for which we calculate only the outgoing edges. We can then apply the Chu-Liu Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967), whose complexity is $O(n^3)$. MST is expected to perform well in the case that there are no additional constraints to be respected. Nonetheless, adding additional constraints is not a trivial matter.

Using the A* algorithm The general schema for A* search has been shown in section 3. Here we detail the heuristics that we have used to guide the search. In A* search, the pending queue is ordered by the estimated cost of a solution whose intermediary state is the current state s. The estimated cost is $f(S) = g(S) + h(S)$ where g is the cost of what decisions have already been taken, here the sum of the cost of selected relations so far (see above). When we build a tree under the right-frontier constraint, we have a set of discourse units yet to be attached. A heuristic yields an optimal solution if it underestimates the remaining cost (it is then an "admissible heuristic"), so that a usual way of estimating this cost is to solve the remaining problem while leaving out some constraints. To be useful the heuristics must also discriminate between comparable states, and be as close as possible to the real cost, so for instance the trivial h(s) = 0, while admissible, is useless. A more classical approach is to consider what would be the best possible decision at a given stage, if no constraint was present. For instance here, we could take, for the estimated cost of attaching a given unit, the best cost of attaching this unit to any other DU already chosen. The remaining cost of the solution is then the sum over the set of remaining DUs. Let's call this h_best. As there can be a lot of variance in the costs, another practical solution is to consider the average of attachments to every remaining nodes. This is potentially not

[5]Two units are adjacent if their distance equals 1.

admissible any more, as it can overestimate the real cost, but can yield good solutions faster. We will call this heuristics h_average. When tested on the development set, the predictions made using h_best and h_average were almost the same so we used h_average in the real experiments because it made decoding faster. Then:

$$f(S) = g(S) + h_{average}(S) = \sum_{(u,v)\in S.E} cost(u,v) + \sum_{u \in S.V} \left(\frac{\sum_{(v \in I/\{u\}} cost(v,u)}{||I||}\right)$$

For each experimental setup, we perform a document-based ten-fold cross-validation[6] on the main part of the corpus.

7 Results

This section reports on the performance obtained for our different systems. Recall that our overall goal is to evaluate labeled discourse structures, encoded here as labeled dependency graphs, produced by the different combinations of local models and decoding strategies. We also have two different training algorithms (NB vs. Maxent) and two settings based on pre-pruning possible attachment points or not. Finally, we are also interested in comparing the joint decoding of the attachments and the relation labels compared to the pipe-lining of the two procedures.

For evaluation, we take the most natural metric for dependency graphs: we compare the set of edges predicted to the reference edges, with precision, recall and F1-measure. We average on the set of all edges on all tested documents (as mentioned in the previous section, we did a cross-validation using 10 document-based folds).

Table 3 presents the results only for attachment of DUs. Besides pruning of the hypothesis space, we tested prediction of attachment alone, and prediction of attachment taking into account the probabilities of the best relations predicted to weight possible attachment (noted "joint unlabeled" evaluation in table 3). Statistical significance was tested by comparing set of scores on documents using Wilcoxon sign-ranked test for paired samples.

Training model	Naive Bayes			Maxent			∅
Decoding method	greedy	MST	A*	greedy	MST	A*	Last
attachment alone (w5)	61.2	65.7	**66.2**	62.1	65.7	65.7	62.4
attachment alone	58.5	62.0	62.1	62.2	*65.7*	*65.7*	62.4
joint/unlabelled (w5)	59.7	61.7	64.8	62.2	65.1	*65.3*	62.4
joint/unlabelled	57.9	57.0	59.6	62.3	65.1	*65.4*	62.4

Table 3: Results for unlabeled structures i.e. attachment of DUs (F1 scores, %). Windowed attachment is marked with (w5). Bold scores are the best overall while italicized scores are the best on a given setup (line). Joint unlabeled evaluation are evaluation of attachments when relations are also used to evaluate the probability of a link between DUs. A* and MST decoding do not differ significantly, but differ from all other methods. Confidence intervals at 95% are all about ± 0.9-1.2% wrt to given scores. Predicting relations does not seem to make a difference for attachment prediction.

[6]Each fold contains instances from a tenth of all documents, every document appearing in only one fold.

We see that A* and MST decoding perform at the same level without significant differences, but largely outperform all other methods. The type of learner used does not seem to make a difference in the pruned version, while Maxent is clearly better when given the whole decision space. This is clearly in line with the extra robustness noted in the classification results in the previous section.

We can observe that the best overall scores are close the F1 score for the pure attachment classification task. This seems to indicate that the impact of our global decoding strategy is not directly captured in the (edge-based) evaluation metric, and is therefore mostly a matter of exhibiting structures with desirable properties, like obeying the RFC. These might in turn be potentially useful for latter processings, such as anaphora resolution.

Training model		Naive Bayes			Maxent			
Decoding method		greedy	MST	A*	greedy	last	MST	A*
joint(w5)	4 rels	38.9	29.3	41.7	42.2	42.2	31.6	*44.1*
joint	4 rels	38.7	26.7	39.6	44.6	44.5	30.0	**46.8**
pipe-line(w5)	4 rels	39.5	42.1	42.5	42.1	42.2	*44.3*	*44.3*
pipe-line	4 rels	38.7	40.8	40.8	44.5	44.5	**46.8**	**46.8**
joint(w5)	18 rels	22.0	8.2	23.7	28.7	28.6	4.8	*30.1*
joint	18 rels	23.4	4.1	24.0	34.2	34.1	5.4	**36.1**
pipe-line(w5)	18 rels	22.5	24.0	24.5	28.7	28.6	*30.2*	*30.2*
pipe-line	18 rels	23.9	24.7	24.8	34.0	34.1	**36.1**	**36.1**

Table 4: Results of for full, labelled structures (F1 scores, %). Windowed attachment is marked with (w5). The 'last' baseline now uses a maxent model for prediction of relations. Bold scores are the best overall while italicized scores are the best on a given setup (line). Confidence intervals at 95% are all about ± 2% wrt to given scores. Best scores on each line are significantly better than the next one at $p<0.01$, except for ties. The best joint and pipe-lined scores are not significantly different from each other.

We now turn to the results on *labeled* discourse structures. The main thing to observe here is that the best decoding methods are still MST and A*, but the (expected) drop induced by relation labeling has confused the attachment results in the case of joint decoding: pipe-lining relation prediction after unlabeled attachment performs significantly better than joint decoding, at least for the best systems. It is also noticeable that pruning the attachment space is not worth it in this configuration (apart from efficiency considerations, of course). The main consequence is that we should refine our relation prediction model before drawing definite conclusions, and extend the approach to larger corpora. In hindsight, fully separating predictions within sentences or between sentences, as done in comparable work, is also something that should have been tried (we only provided features to that effect).

How can we compare these results to similar work? Taking directly Hernault et al. (2010) published scores is not easy, since they produce RST constituent trees, and use dedicated measures, namely the Parseval measures, which compares common subtrees. That is why we tried to reproduced their overall method of decoding on our data (the greedy procedure). We can still have a look at their classification scores for attachments, and the range of their evaluation for the whole structure. In their framework, it is equivalent to the tasks they call

"structure" and "nuclearity", the first being finding a pair to link, the second one being the choice of direction of the attachment, choosing the "head" of the result. When using perfect segmentation, as we do, (Hernault et al., 2010) have a F1 score of 68.4%. In (Subba and Di Eugenio, 2009), structure is predicted as relation argument spans with a F1 score of 70%, but this is similar to the attach-to-last baseline, and nuclearity is at 50% (the same baseline is at 48). It is also interesting to consider simple attachment predictions as made in (Feng and Hirst, 2012). Without knowledge of the rest of the structure, they have a F1 score of 69.84% on attachment decisions.

Again, it is hard to compare, since we don't have information about baselines comparable to ours, but we can observe the scores are in the same close ranges, while we have a simpler model with less features, trained on much less instances. Hernault et al. (2010) have 17k positively attached pairs, 77k overall, for a ratio of 23%, while we have, in the pruned version, 2.5k positive instances out of 13k (ratio $\leq$20%), and 2.7k out of 125k for the full version (2%).

For labelled structured, (Subba and Di Eugenio, 2009) report a F1 score of 35%, with 15 relations and a baseline of 22%, slightly less than our 18-relation model, while the accuracy of their relation labeler reaches 60% (a much better score than ours).

Finally, Baldridge and Lascarides (2005), who use 30 relations on the very specialized corpus of Verbmobil dialogues, report F1 scores of 68% for unlabeled structures (very close to dependency graphs), and 43% for labelled structures. The task is arguably easier since dialogues are more constrained, and since they manually annotated some features used by their probabilistic model (e.g. semantic tags, utterance mood).

Conclusions

We have proposed a general approach to discourse structure prediction at the document level, performing a global search over the space of possible structures while optimizing a global criterion over the set of potential coherence relations, in order to take into account linguistically motivated constraints on the predicted structure (e.g. the RFC). We tested this approach on a corpus of French texts which assumes theoretical aspects from SDRT, but it could and will be adapted to other type of discourse annotations, such as RST corpora, using a simple algorithm alluded to above for translating RST into dependency graphs. The results for our general approach are at least comparable to similar approaches which are arguably more specific to a given corpus and have significantly more data to train. Our relation prediction model is slightly less accurate, being induced from a smaller dataset and poorer features than the best models, but the global decoding improvements are significant over other decoding approaches, while the predictions respect the desired properties on discourse structures. Besides improving our relation model, we intend to separate completely the predictions of links for intra-sentential discourse units from other relations, as many approaches have shown that sub-problem to be much easier. We also intend to have a more structured approach to learning the structures, first by integrating decoding within the learning phase, as is done in the syntactic analysis literature we took inspiration from, and secondly by studying the usefulness of k-best parsing on the overall result.

Acknowledgments

We wish to thank our colleagues M. Serrurier and J. Mengin.

References

Afantenos, S., Asher, N., Benamara, F., Bras, M., Fabre, C., Ho-Dac, M., Draoulec, A. L., Muller, P., Pery-Woodley, M.-P., Prevot, L., Rebeyrolles, J., Tanguy, L., Vergez-Couret, M., and Vieu, L. (2012). An empirical resource for discovering cognitive principles of discourse organisation: the annodis corpus. In Calzolari, N., Choukri, K., Declerck, T., Doğan, M. U., Maegaard, B., Mariani, J., Odijk, J., and Piperidis, S., editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Asher, N. and Lascarides, A. (2003). *Logics of Conversation*. Studies in Natural Language Processing. Cambridge University Press, Cambridge, UK.

Baldridge, J., Asher, N., and Hunter, J. (2007). Annotation for and Robust Parsing of Discourse Structure on Unrestricted Texts. *Zeitschrift fur Sprachwissenschaft*, 26:213–239.

Baldridge, J. and Lascarides, A. (2005). Probabilistic head-driven parsing for discourse structure. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL)*.

Carlson, L., Marcu, D., and Okurowski, M. E. (2003). Building a discourse-tagged corpus in the framework of rhetorical structure theory. In van Kuppevelt, J. and Smith, R., editors, *Current Directions in Discourse and Dialogue*, pages 85–112. Kluwer Academic Publishers.

Chu, Y. J. and Liu, T. H. (1965). On the shortest arborescence of a directed graph. *Science Sinica*, 14:1396–1400.

Curk, T., Demšar, J., Xu, Q., Leban, G., Petrovič, U., Bratko, I., Shaulsky, G., and Zupan, B. (2005). Microarray data mining with visual programming. *Bioinformatics*, 21:396–398.

Daumé III, H. (2004). Notes on CG and LM-BFGS optimization of logistic regression. Paper available at `http://pub.hal3.name#daume04cg-bfgs`, implementation available at `http://hal3.name/megam/`.

duVerle, D. and Prendinger, H. (2009). A novel discourse parser based on support vector machine classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 665–673, Suntec, Singapore. Association for Computational Linguistics.

Edmonds, J. (1967). Optimum branchings. *Journal of Research of the National Bureau of Standards*, 71B(233–240).

Feng, V. W. and Hirst, G. (2012). Text-level discourse parsing with rich linguistic features. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 60–68, Jeju Island, Korea. Association for Computational Linguistics.

Gerber, M., Gordon, A., and Sagae, K. (2010). Open-domain commonsense reasoning using discourse relations from a corpus of weblog stories. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 43–51, Los Angeles, California. Association for Computational Linguistics.

Hernault, H., Prendinger, H., duVerle, D. A., and Ishizuka, M. (2010). HILDA: A Discourse Parser Using Support Vector Machine Classification. *Dialogue and Discourse*, 1(3):1–33.

Joty, S., Carenini, G., and Ng, R. (2012). A novel discriminative framework for sentence-level discourse analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 904–915, Jeju Island, Korea. Association for Computational Linguistics.

Lin, Z., Kan, M.-Y., and Ng, H. T. (2009). Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 343–351, Singapore. Association for Computational Linguistics.

Lin, Z., Ng, H. T., and Kan, M.-Y. (2011). Automatically evaluating text coherence using discourse relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 997–1006, Portland, Oregon, USA. Association for Computational Linguistics.

Louis, A., Joshi, A., Prasad, R., and Nenkova, A. (2010). Using entity features to classify implicit discourse relations. In *Proceedings of the SIGDIAL 2010 Conference*, pages 59–62, Tokyo, Japan. Association for Computational Linguistics.

Mann, W. C. and Thompson, S. A. (1987). Rhetorical Structure Theory: A Framework for the Analysis of Texts. Technical Report ISI/RS-87-185, Information Sciences Institute, Marina del Rey, California.

Marcu, D. (2000). *The Theory and Practice of Discourse Parsing and Summarization*. The MIT Press.

Marcu, D. and Echihabi, A. (2002). An unsupervised approach to recognizing discourse relations. In *Proceedings of ACL*, pages 368–375.

McDonald, R. T., Pereira, F., Ribarov, K., and Hajic, J. (2005). Non-projective dependency parsing using spanning tree algorithms. In *HLT/EMNLP*.

Pauls, A. and Klein, D. (2009). K-best a* parsing. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 958–966, Suntec, Singapore. Association for Computational Linguistics.

Polanyi, L. (1988). A formal model of the structure of discourse. *Journal of Pragmatics*, 12:601–638.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. L. (2008). The Penn Discourse TreeBank 2.0. In *Proceedings of LREC 2008*.

Sagae, K. (2009). Analysis of discourse structure with syntactic dependencies and data-driven shift-reduce parsing. In *Proceedings of the 11th International Conference on Parsing Technologies*, IWPT '09, pages 81–84, Stroudsburg, PA, USA. Association for Computational Linguistics.

Somasundaran, S., Namata, G., Wiebe, J., and Getoor, L. (2009). Supervised and unsupervised methods in employing discourse relations for improving opinion polarity classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 170–179, Singapore. Association for Computational Linguistics.

Soricut, R. and Marcu, D. (2003). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 149–156. Association for Computational Linguistics.

Sporleder, C. and Lascarides, A. (2005). Exploiting linguistic cues to classify rhetorical relations. In *Proceedings of Recent Advances in Natural Langauge Processing (RANLP)*, Bulgaria.

Sporleder, C. and Lascarides, A. (2008). Using automatically labelled examples to classify rhetorical relations: An assessment. *Natural Language Engineering*, 14(3):369–416.

Stede, M. (2011). *Discourse Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Subba, R. and Di Eugenio, B. (2009). An effective discourse parser that uses rich linguistic information. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 566–574, Boulder, Colorado. Association for Computational Linguistics.

Verberne, S., Boves, L., Oostdijk, N., and Coppen, P.-A. (2007). Evaluating discourse-based answer extraction for why-question answering. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 735–736, New York, NY, USA. ACM.

Wellner, B., Pustejovsky, J., Havasi, C., Rumshisky, A., and Saurí, R. (2006). Classification of discourse coherence relations: an exploratory study using multiple knowledge sources. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, SigDIAL '06, pages 117–125, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wolf, F. and Gibson, E. (2006). *Coherence in Natural Language: Data Stuctures and Applications*. The MIT Press.