Integer Linear Programming for Discourse Parsing

Jérémy Perret Stergos Afantenos Nicholas Asher Mathieu Morey
IRIT, Université de Toulouse & CNRS
118 Route de Narbonne, 31062 Toulouse, France
{firstname.lastname@irit.fr}

Abstract

In this paper we present the first, to the best of our knowledge, discourse parser that is able to predict non-tree DAG structures. We use Integer Linear Programming (ILP) to encode both the objective function and the constraints as global decoding over local scores. Our underlying data come from multi-party chat dialogues, which require the prediction of DAGs. We use the dependency parsing paradigm, as has been done in the past (Muller et al., 2012; Li et al., 2014; Afantenos et al., 2015), but we use the underlying formal framework of SDRT and exploit SDRT's notions of *left* and *right distributive* relations. We achieve an F-measure of 0.531 for fully labeled structures which beats the previous state of the art.

1 Introduction

Multi-party dialogue parsing, in which complete discourse structures for multi-party dialogue or its close cousin, multi-party chat, are automatically constructed, is still in its infancy. Nevertheless, these are now very common forms of communication on the Web. Dialogue appears also importantly different from monologue. Afantenos et al. (2015) point out that forcing discourse structures to be trees will perforce miss 9% of the links in their corpus, because a significant number of discourse structures in the corpus are not trees. Although Afantenos et al. (2015) is the only prior paper we know of that studies dialogue parsing on multi-party dialogue, and that work relied on methods adapted to treelike structures, we think the area of multi-party dialogue and non-treelike discourse structures is ripe for investigation and potentially important for other genres like the discourse analysis of fora (Wang et al., 2011, for example). This paper proposes a method based on constraints using Integer Linear Programming decoding over local probability distributions to investigate both treelike and non-treelike, full discourse structures for multi-party dialogue. We show that our method outperforms that of Afantenos et al. (2015) on the corpus they developed.

Discourse parsing involves at least three main steps: the segmentation of a text into *elementary discourse units* (EDUs), the basic building blocks for discourse structures, the attachment of EDUs together into connected structures for texts, and finally the labelling of the links between discourse units with discourse relations. Much current work in discourse parsing focuses on the labelling of discourse relations, using data from the Penn Discourse Treebank (PDTB) (Prasad et al., 2008). This work has availed itself of increasingly sophisticated features of the semantics of the units to be related (Braud and Denis, 2015); but as the PDTB does not provide full discourse structures for texts, it is not relevant to our concerns here. *Rhetorical Structure Theory* (RST) (Mann and Thompson, 1987; Mann and Thompson, 1988; Taboada and Mann, 2006) does take into account the global structure of the document, and the RST Discourse Tree Bank Carlson et al. (2003) has texts annotated according to RST with full discourse structures. This has guided most work in recent discourse parsing of multi-sentence text (Subba and Di Eugenio, 2009; Hernault et al., 2010; duVerle and Prendinger, 2009; Joty et al., 2013; Joty et al., 2015).

Proceedings of NAACL-HLT 2016, pages 99–109,
San Diego, California, June 12-17, 2016.

But RST requires that discourse structures be projective trees.

While projective trees are arguably a contender for representing the discourse structure of monologue text, multi-party chat dialogues exhibit crossing dependencies. This rules out using a theory like RST as a basis either for an annotation model or as a guide to learning discourse structure (Afantenos et al., 2015). Several subgroups of interlocutors can momentarily form and carry on a discussion amongst themselves, forming thus multiple concurrent discussion threads. Furthermore, participants of one thread may reply or comment to something said to another thread. One might conclude from the presence of multiple threads in dialogue that we should use non-projective trees to guide discourse parsing. But non-projective trees cannot always reflect the structure of a discourse either, as Asher and Lascarides (2003) argue on theoretical grounds. Afantenos et al. (2015) provide examples in which a question or a comment by speaker S that is addressed to all the engaged parties in the conversation receives an answer from all the other participants, all of which are then acknowledged by S with a simple *OK* or *No worries*, thus creating an intuitive, "lozenge" like structure, in which the acknowledgment has several incoming links representing discourse dependencies.

A final, important organizing element of the discourse structure for text and dialogue is the presence of clusters of EDUs that can act together as an argument to other discourse relations. This means that subgraphs of the entire discourse graph act as elements or nodes in the full discourse structure. These subgraphs are *complex discourse units* or CDUs.[1] Here is an example from the *Settlers* corpus:

(1) gotwoodforsheep: [Do you have a sheep?]$_a$
Thomas: [I do,]$_b$ [if you give me clay]$_c$
Thomas: [or wood.]$_d$

Thomas's response to gotwoodforsheep spans two turns in the corpus. More interestingly, the response is a conditional "yes" in which EDUs (c) and (d) jointly specify the antecedent of the discourse relation that links both to the EDU *I do*.

[1]CDUs are a feature of SDRT as we explain below. They are also a feature of RST on some interpretations of the Satellite-Nucleus feature.

CDUs have been claimed to be an important organizing principle of discourse structure and important for the analysis of anaphora and ellipsis for over 20 years (Asher, 1993). Yet the computational community has ignored them; when they are present in annotated corpora, they have been eliminated. This attitude is understandable, because CDUs, as they stand, are not representable as trees in any straightforward way. But given that our method can produce non-treelike graphs, we take a first step towards the prediction of CDUs as part of discourse structure by encoding them in a hypergraph-like framework. In particular, we will transform our corpus by distributing relations on CDUs over all their constituents as we describe in section 3.

Our paper is organized as follows. The data that we have used are described in more detail in the following section, while the underlying linguistic theory that we are using is described in section 3. In section 4 we present in detail the model that we have used, in particular the ILP decoder and the constraints and objective function it exploits. We report our results in section 5. Section 6 provides the related work while section 7 concludes this paper.

2 Input data

For our experiments we used a corpus collected from chats involving an online version of the game *The Settlers of Catan* described in (Afantenos et al., 2012; Afantenos et al., 2015). *Settlers* is a multiparty, win-lose game in which players use resources such as wood and sheep to build roads and settlements. Players take turns directing the bargaining. This is the only discourse annotated corpus of multiagent dialogue of which we are aware, and it was one in which apparently non-treelike structures were already noted and also contains CDUs. Such a chat corpus is also useful to study because it approximates spoken dialogue in several ways—sentence fragments, non-standard orthography and occasional lack of syntax—without the inconvenience of transcribing speech. The corpus consists of 39 games annotated for discourse structure in the style of SDRT. Each game consists of several dialogues, and each dialogue represents a single bargaining session directed by a particular player or perhaps several connected sessions. Each dialogue is treated as hav-

	Total	Training	Testing
Dialogues	1091	968	123
Turns	9160	8166	994
EDUs	10677	9545	1132
CDUs	1284	1132	152
Relation instances			
No distribution	10191	9127	1064
Partial dist.	11734	10507	1227
Full dist.	13675	12210	1465

Table 1: Dataset overview

ing its own discourse structure. About 10% of the corpus was held out for evaluation purposes while the rest was used for training. The dialogues in the corpus are mostly short with each speaker's turn containing typically only one, two or three EDUs, though the longest has 156 EDUs and 119 turns. Most of the discourse connections or relation instances in the corpus thus occur between speaker turns. Statistics on the number of dialogues, EDUs and relations contained in each sub-corpus can be found in table 1. Note that the number of relation instances in the corpus depends on how CDUs are translated, which we'll explain in the next section. The corpus has approximately the same number of EDUs and relations as the RST corpus (Carlson et al., 2003).

3 Linguistic Foundations

Segmented Discourse Representation Theory. We give a few details here on one discourse theory in which non-treelike discourse structures are countenanced and that underlies the annotations of the corpus we used. That theory is SDRT. In SDRT, a discourse structure, or *SDRS*, consists of a set of Discourse Units (DUs) and as Discourse Relations linking those units. DUs are distinguished into EDUs and CDUs. We identify EDUs here with phrases or sentences describing a state or an event; CDUs are SDRSs. Formally an SDRS for a given text segmented in EDUs $D = \{e_1, \ldots, e_n\}$, where e_i are the EDUs of D, is a tuple (V, E_1, E_2, ℓ) where V is a set of nodes or discourse units including $\{e_1, \ldots, e_n\}$, $E_1 \subseteq V \times V$ a set of edges representing discourse relations, $E_2 \subseteq V \times V$ a set of edges that represents parthood in the sense that if $(x, y) \in E_2$, then the unit x is an element of the CDU y; finally ℓ: $E_1 \rightarrow$ *Relations* is a labelling function that assigns an edge in E_1 its discourse relation type.

From SDRT Structures to Dependency Structures. Predicting full SDRSs (V, E_1, E_2, ℓ) with $E_2 \neq \varnothing$ has been to date impossible, because no reliable method has been identified in the literature for calculating edges in E_2. Instead, most approaches (Muller et al., 2012; Afantenos et al., 2015, for example) simplify the underlying structures by a *head replacement strategy (HR)* that removes nodes representing CDUs from the original hypergraphs and replacing any incoming or outgoing edges on these nodes on the *heads* of those CDUs, forming thus dependency structures and not hypergraphs. A similar approach has also been followed by Hirao et al. (2013) and Li et al. (2014) in the context of RST to deal with multi-nuclear relations.

Transforming SDRSs using HR does not come without its problems. The decision to attach all incoming and outgoing links to a CDU to its head is one with little theoretical or semantic justification. The semantic effects of attaching an EDU to a CDU are not at all the same as attaching an EDU to the head of the CDU. For example, suppose we have a simple discourse with the following EDUs marked by brackets and discourse connectors in bold :

(2) [The French economy continues to suffer]$_a$ **because** [high labor costs remain high]$_b$ **and** [investor confidence remains low]$_c$.

The correct SDRS for (2) is one in which both b and c *together* explain why the French economy continues to suffer. That is, b and c form a CDU and give rise to the following graph:

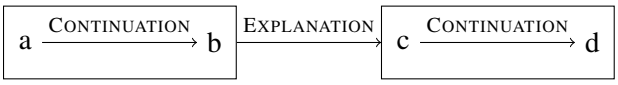
$$a \xrightarrow{\text{EXPLANATION}} \boxed{b \xrightarrow{\text{CONTINUATION}} c}$$

HR on (2) produces a graph whose strictly compositional interpretation would be false—b alone explains why the French economy continues to suffer. Alternatively an interpretation of the proposed translation an SDRS with CDUs would introduce spurious ambiguities: either b alone or b and c together provide the explanation. To make matters worse, given the semantics of discourse relations

in SDRT (Asher and Lascarides, 2003), some relations have a semantics that implies that a relation between a CDU and some other discourse unit can be distributed over the discourse units that make up the CDU. But not all relations are distributive in this sense. For example, we could complicate (2) slightly:

(3) [The French economy continues to suffer]$_a$ **and** [the Italian economy remains in the doldrums]$_b$ **because of** [persistent high labor costs]$_c$ **and** [lack of investor confidence in both countries]$_d$.

In (3), the SDRS graph would be:



However, this SDRS entails that a is explained by $[c, d]$ *and* that b is explained by $[c, d]$. That is, EXPLANATION "distributes" to the left but not to the right. Once again, the HR translation from SDRSs into dependency structures described above would get the intuitive meaning of this example wrong or introduce spurious ambiguities.

Given the above observations, we decided to take into account the formal semantics of the discourse relations before replacing CDUs. More precisely, we distinguish between *left distributive* and *right distributive* relations. In a nutshell, we examined the temporal and modal semantics of relations and classified them as to whether they were distributive with respect to their left or to their right argument; left distributive relations are those for which the source CDU node should be distributed while right distributive relations are those for which the target CDU node should be distributed. A relation can be both left and right distributive. Left distributive relations include ACKNOWLEDGEMENT, EXPLANATION, COMMENT, CONTINUATION, NARRATION, CONTRAST, PARALLEL, BACKGROUND, while right distributive relations include RESULT, CONTINUATION, NARRATION, COMMENT, CONTRAST, PARALLEL, BACKGROUND, ELABORATION. In Figure 1 we show an example of how relations distribute between EDU/CDU, CDU/EDU and CDU/CDU.

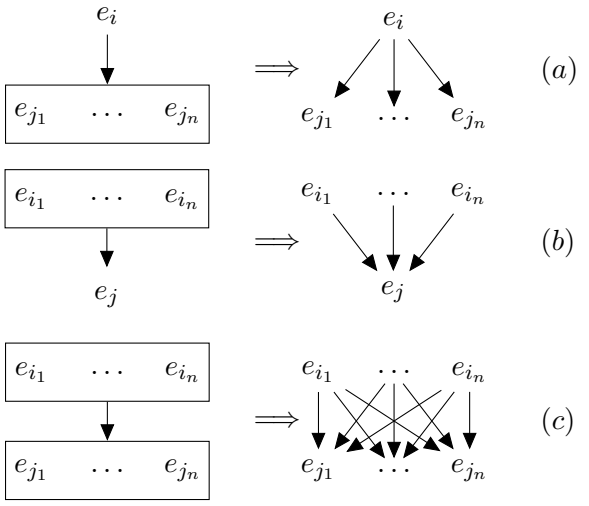


Figure 1: Distributing relations: (a) *right* distribution from an EDU to a CDU, (b) *left* distribution from a CDU to an EDU, (c) from a CDU to a CDU. We assume that all relations are both *right* and *left* distributive.

4 Underlying Model

Decoding over local scores. When we apply either a full or partial distributional (partial distribution takes into account which relations distribute in which direction) translation to the SDRSs in our corpus, we get dependency graphs that are not trees as input to our algorithms. We now approximate full SDRS graphs (V, E_1, E_2, ℓ) with graphs that distribute out E_2—that is, graphs of the form (V, E_1, ℓ) or more simply (V, E, ℓ). It is important to note that those graphs are not in general trees but rather Directed Acyclic Graphs (DAGs). We now proceed to detail how we learn such structures.

Ideally, what one wants is to learn a function $h : \mathcal{X}_{E^n} \mapsto \mathcal{Y}_\mathcal{G}$ where $\mathcal{X}_{E^n}$ is the domain of instances representing a collection of EDUs for each dialogue and $\mathcal{Y}_\mathcal{G}$ is the set of all possible SDRT graphs. However, given the complexity of this task and the fact that it would require an amount of training data that we currently lack in the community, we aim at the more modest goal of learning a function $h : \mathcal{X}_{E^2} \mapsto \mathcal{Y}_R$ where the domain of instances $\mathcal{X}_{E^2}$ represents parameters for a pair of EDUs and $\mathcal{Y}_R$ represents the set of SDRT relations.

An important drawback of this approach is that there are no formal guarantees that the predicted structures will be well-formed. They could for ex-

ample contain cycles although they should be DAGs. Most approaches have circumvented this problem by using global decoding over local scores and by imposing specific constraints upon decoding. But, those constraints were mostly limited to the production of maximum spanning trees, and not full DAGs. We perform global decoding as well but use Integer Linear Programming (ILP) with an objective function and constraints that allow non-tree DAGs. We use a regularized maximum entropy (shortened MaxEnt) model (Berger et al., 1996) to get the local scores, both for attachment and labelling.

ILP for Global Decoding. ILP essentially involves an objective function that needs to be maximized under specific constraints. Our goal is to build the directed graph $G = \langle V, E, R \rangle$ with R being a function that provides labels for the edges in E. Vertices (EDUs) are referred by their position in textual order, indexed from 1. The m labels are referred by their index in alphabetical order, starting from 1. Let $n = |V|$.

The local model provides us with two real-valued functions:

$$s_a : \{1, \ldots, n\}^2 \mapsto [0, 1]$$
$$s_r : \{1, \ldots, n\}^2 \times \{1, \ldots, m\} \mapsto [0, 1]$$

$s_a(i, j)$ gives the score of attachment for a pair of EDUs (i, j); $s_r(i, j, k)$ gives the score for the attached pair of EDUs (i, j) linked with the relation type k. We define the n^2 binary variables a_{ij} and mn^2 binary variables r_{ijk}:

$$a_{ij} = 1 \equiv (i, j) \in V$$
$$r_{ijk} = 1 \equiv R(i, j) = k$$

The objective function that we want to maximize is

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \left(a_{ij} s_a(i, j) + \sum_{k=1}^{m} r_{ijk} s_r(i, j, k) \right)$$

which gives us a score and a ranking for all candidate structures.

Our objective function is subject to several constraints. Because we have left the domain of trees well-explored by syntactic analysis and their computational implementations, we must design new constraints on discourse graphs, which we have developed from looking at our corpus while also being guided by theoretical principles. Some of these constraints come from SDRT, the underlying theory of the annotations. In SDRT discourse graphs should be DAGs with a unique root or source vertex, i.e. one that has no incoming edges. They should also be weakly connected; i.e. every discourse unit in it is connected to some other discourse unit. We implemented connectedness and the unique root property as constraints in ILP by using the following equations.

$$\sum_{i=1}^{n} h_i = 1$$
$$\forall j \quad 1 \leq n h_j + \sum_{i=1}^{n} a_{ij} \leq n$$

where h_i is a set of auxiliary variables indexed on $\{1, \ldots, n\}$. The above constraint presupposes that our graphs are acyclic.

Implementing acyclicity is facilitated by another constraint that we call the *turn constraint*. This constraint is also theoretically motivated. The graphs in our training corpus are *reactive* in the sense that speakers' contributions are reactions and attach anaphorically to prior contributions of other speakers. This means that edges between the contributions of different speakers are always oriented in one direction. A turn by one speaker can't be anaphorically and rhetorically dependent on a turn by another speaker that comes after it. Once made explicit, this constraint has an obvious rationale: people do not know what another speaker will subsequently say and thus they cannot create an anaphoric or rhetorical dependency on this unknown future act. This is not the case within a single speaker turn though; people can know what they will say several EDUs ahead so they can make such kinds of future directed dependencies. ILP allows us to encode this constraint as follows. We indexed turns from different speakers in textual order from 1 to n_t, while consecutive turns from the same speaker were assigned the same index. Let $t(i)$ be the turn index of EDU i, and $T(k)$ the set of all EDUs belonging to turn k. The following constraint forbids backward links between EDUs from distinct turns:

$$\forall i, j \quad (i > j) \wedge (t(i) \neq t(j)) \implies a_{ij} = 0$$

The observation concerning the turn constraint is also useful for the model that provides local scores. We used it for attachment and relation labelling during training and testing.

Given the turn constraint we only need to ensure acyclicity of the same speaker turn subgraphs. We introduce an auxiliary set of integer variables, (c_{ki}), indexed on $\{1, \ldots, n_t\} \times \{1, \ldots, n\}$ in order to express this constraint:

$$\forall k, i \quad 1 \leq c_{ki} \leq |T(k)|$$
$$\forall k, i, j \text{ such that } t(i) = t(j) = k$$
$$c_{kj} \leq c_{ki} - 1 + n(1 - a_{ij})$$

Another interesting observation concerns the density of the graph. The objective function being additive on positive terms, every extra edge improves the global score of the graph, which leads to an almost-complete graph unless the edge count is constrained. So we imposed an upper limit $\delta \in [1, n]$ representing the density of the graphs:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \leq \delta(n - 1)$$

$\delta \in [1, n]$ since we need to have at least $n - 1$ edges for the graph to be connected and at maximum we can have $n(n - 1)$ edges if the graph is complete without loops. δ being a hyper-parameter, we estimated it on a development corpus representing 20% of our total corpus.[2]

The development corpus also shows that graph density decreases as the number of vertices grow. A high δ entails a too large number of edges in longer dialogues. We compensate for this effect by using an additive cap $\eta \geq 0$ on the edge count, also estimated on the development corpus:[3]

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \leq n - 1 + \eta$$

Another empirical observation concerning the corpus was that the number of outgoing edges from any EDU had an upper bound $e_o \ll n$. We set that as an ILP constraint:[4]

$$\forall i \quad \sum_{j=1}^{n} a_{ij} \leq e_o$$

These observations don't have a semantic explanation, but they suggest a pragmatic one linked at least to the type of conversation present in our corpus. Short dialogues typically involve a opening question broadcast to all the players in search of a bargain, and typically all the other players reply. The replies are then taken up and either a bargain is reached or it isn't. The players then move on. Thus, the density of the graph in such short dialogues will be determined by the number of players (in our case, four). In a longer dialogue, we have more directed discourse moves and threads involving subgroups of the participants appear, but once again in these dialogues it never happens that our participants return again and again to the same contribution; if the thread of commenting on a contribution ϕ continues, future comments attach to prior comments, not to ϕ. Our ILP constraints on density and edge counts thus suggest a novel way of capturing different dialogue types and linguistic constraints.

Finally, we included various minor constraints, such as the fact that EDUs cannot be attached to themselves,[5] if EDUs i and j are not attached the pair is not assigned any discourse relation label,[6] EDUs within a sequence of contributions by the same speaker in our corpus are linked at least to the previous EDU (Afantenos et al., 2015)[7] and edges with zero score are not included in the graph.[8]

For purposes of comparison with the ILP decoder, we tested the Chu-Liu-Edmonds version of the classic Maximum Spanning Tree (MST) algorithm McDonald et al. (2005) used for discourse parsing by Muller et al. (2012) and Li et al. (2014) and by Afantenos et al. (2015) on the *Settlers* corpus. This algorithm requires a specific node to be the root, i.e. a node without any incoming edges, of the initial complete graph. For each dialogue, we made an artificial node as the root with special dummy features. At the end of the procedure, this node points to the real root of the discourse graph. As baseline measures, we included what we call a LOCAL decoder which creates a simple classifier out of the raw local probability distribution. Since we use MaxEnt, this

[2] δ takes the values 1.0, 1.2 and 1.4 for the head, partial and full distribution of the relations, respectively.

[3] η takes the value of 4 for the full distribution while it has no upper bound for the head and partial distributions.

[4] e_o is estimated on the development corpus to the value of 6 for the head, partial and full distributions.

[5] $\forall i \; a_{ii} = 0$

[6] $\forall i, j \; \sum_{k=1}^{m} r_{ijk} = a_{ij}$

[7] $\forall i \quad t(i) = t(i + 1) \implies a_{i,i+1} = 1$

[8] $\forall i, j \quad s_a(i, j) = 0 \implies a_{ij} = 0$ and $\forall i, j, k \quad s_r(i, j, k) = 0 \implies x_{ijk} = 0$

decoder selects

$$\hat{r} = \underset{r}{\operatorname{argmax}} \left(\frac{1}{Z(c)} \exp \left(\sum_{i=1}^{m} w_i f_i(p, r) \right) \right)$$

with r representing a relation type or a binary attachment value. A final baseline was LAST, where each EDU is attached to the immediately preceding EDU in the linear, textual order.

5 Experiments and Results

Features for training the local model and getting scores for the decoders were extracted for every pair of EDUs. Features concerned each EDU individually as well as the pair itself. We used obvious, surface features such as: the position of EDUs in the dialogue, who their speakers are, whether two EDUs have the same speaker, the distance between EDUs, the presence of mood indicators ('?', '!') in the EDU, lexical features of the EDU (e.g., does a verb signifying an exchange occur in the EDU), and first and last words of the EDU. We also used the structures and Subject lemmas given by syntactic dependency parsing, provided by the Stanford CoreNLP pipeline (Manning et al., 2014). Finally we used Cadilhac et al. (2013)'s method for classifying EDUs with respect to whether they involved an offer, a counteroffer, or were other.

As mentioned earlier, in addition to the ILP and MST decoders we used two baseline decoders, LAST and LOCAL. The LAST decoder simply selects the previous EDU for attachment no matter what the underlying probability distribution is. This has proved a very hard baseline to beat in discourse. The LOCAL decoder is a naive decoder which in the case of attachment returns "attached" if the probability of attachment between EDUs i and j is higher than .5 and "non-attached" in the opposite case.

Each of the three distribution methods described in Section 3 (Head, Partial and Full Distribution) yielded different dependency graphs for our input documents, which formed three distinct corpora on which we trained and tested separately. For each of them, our training set represented 90% of the dependency graphs from the initial corpus, chosen at random; the test set representing the remaining 10%. Table 2 shows our evaluation results, comparing decoders and baselines for each of the distribution strategies. As can be seen, our ILP decoder consistently performs significantly better than the baselines as well as the MST decoder, which was the previous state of the art (Afantenos et al., 2015) even when restricted to tree structures and HR (setting the hyper-parameter $\delta = 1$). This prompted us to investigate how our objective function compared to MST's. We eliminated all constraints in ILP except acyclicity, connectedness, turn constraint and eliminating any constraint on outgoing edges (setting $\delta = \infty$); in this case, ILP's objective function performed better on the full structure prediction (.531 F1) than MST with attachment and labelling jointly maximized (.516 F1). This means that our objective function, although it maximizes scores and not probabilities, produces an ordering over outputs that outperforms classic MST. Our analysis showed further that the constraints on outgoing edges (the tuning of the hyperparameter $e_o = 6$) were very important for our corpus and our (admittedly flawed) local model; in other words, an ILP constrained tree for this corpus was a better predictor of the data with our local model than an unrestrained MST tree decoding.

We also note that our scores dropped in distributive settings but that ILP performed considerably better than the alternatives and better than the previous state of the art on dependency trees using HR on the gold and MST decoding. We need to investigate further constraints, and to refine and improve our features to get a better local model. Our local model will eventually need to be replaced by one that takes into account more of the surrounding structure when it assigns scores to attachments and labels. We also plan to investigate the use of recurrent neural networks in order to improve our local model.

6 Related Work

ILP has been used for various computational linguistics tasks: syntactic parsing (Martins et al., 2010; Fernández-González and Martins, 2015), semantic parsing (Das et al., 2014), coreference resolution (Denis and Baldridge, 2007) and temporal analysis (Denis and Muller, 2011). As far as we know, we are the first to use ILP to predict discourse structures.

Our use of dependency structures for discourse also has antecedents in the literature. The first we know of is Muller et al. (2012). Their prediction

Decoder	Model	Unlabelled Attachment			Labelled Attachment		
		Precision	Recall	F1	Precision	Recall	F1
Head (no distribution)							
LAST	–	0.602	0.566	0.584	0.403	0.379	0.391
LOCAL	local	0.664	0.379	0.483	0.591	0.337	0.429
MST	local	0.688	0.655	0.671	0.529	0.503	0.516
ILP	local	0.707	0.672	**0.689**	0.544	0.518	**0.531**
Partial distribution							
LAST	–	0.651	0.545	0.593	0.467	0.391	0.426
LOCAL	local	0.647	0.370	0.471	0.544	0.311	0.396
MST	local	0.710	0.594	0.647	0.535	0.448	0.488
ILP	local	0.680	0.657	**0.668**	0.528	0.510	**0.519**
Full distribution							
LAST	–	0.701	0.498	0.582	0.505	0.360	0.420
LOCAL	local	0.681	0.448	0.541	0.558	0.367	0.443
MST	local	0.737	0.524	0.613	0.561	0.399	0.466
ILP	local	0.703	0.649	**0.675**	0.549	0.507	**0.527**

Table 2: Evaluation results.

model uses local probability distributions and global decoding, and they transform their data using HR, and so ignore the semantics of discourse relations. Hirao et al. (2013) and Li et al. (2014) also exploit dependency structures by transforming RST trees. Li et al. (2014) used both the Eisner algorithm (Eisner, 1996) as well as the MST algorithm as decoders. We plan to apply ILP techniques to the RST Tree Bank to compare our method with theirs.

Most work on discourse parsing focuses on the task of discourse relation labeling between pairs of discourse units—e.g., Marcu and Echihabi (2002) Sporleder and Lascarides (2005) and Lin et al. (2009)—without worrying about global structure. In essence the problem that they treat corresponds only to our local model. As we have argued above, this setting makes an unwarranted assumption, as it assumes independence of local attachment decisions. There is also work on discourse structure within a single sentence; e.g., Soricut and Marcu (2003), Sagae (2009). Such approaches do not apply to our data, as most of the structure in our dialogues lies beyond the sentence level.

As for other document-level discourse parsers, Subba and Di Eugenio (2009) use a transition-based approach, following the paradigm of Sagae (2009). duVerle and Prendinger (2009) and Hernault et al. (2010) both rely on locally greedy methods. They treat attachment prediction and relation label prediction as independent problems. Feng and Hirst (2012) extend this approach by additional feature engineering but is restricted to sentence-level parsing. Joty et al. (2013) and Joty et al. (2015) present a text-level discourse parser that uses Conditional Random Fields to capture label inter-dependencies and chart parsing for decoding and have the best results on non-dependency based discourse parsing, with an F1 of 0.689 on unlabelled structures and 0.5587 on labelled structures.

The afore-cited work concerns only monologue. Baldridge and Lascarides (2005) predicted tree discourse structures for 2 party "directed" dialogues from the Verbmobil corpus by training a PCFG that exploited the structure of the underlying task. Elsner and Charniak (2010), Elsner and Charniak (2011) present a combination of local coherence models initially provided for monologues showing that those models can satisfactorily model local coherence in chat dialogues. However, they do not present a full discourse parsing model. Our data required a more open domain approach and a more sophisticated approach to structure. Afantenos et al. (2015) worked on multi-party chat dialogues with the same corpus, but they too did not consider the semantics of discourse relations and replaced CDUs with their heads using HR. While this allowed them to

use MST decoding over local probability distributions, this meant that their implementation had inherent limitations because it is limited to producing tree structures. They also used the turn constraint, but imposed exogenously to decoding; ILP allows us to integrate it into the structural decoding. We achieve better results than they on treelike graphs and we can explore the full range of non-treelike discourse graphs within the ILP framework. Our parser has thus much more room to improve than those restricted to MST decoding.

7 Conclusions and future work

We have presented a novel method for discourse parsing of multiparty dialogue using ILP with linguistically and empirically motivated constraints and an objective function that integrates both attachment and labelling tasks. We have shown also that our method performs better than the competition on multiparty dialogue data and that it can capture non-treelike structures found in the data.

We also have a better treatment of the hierarchical structure of discourse than the competition. Our treatment of CDUs in discourse annotations proposes a new distributional translation of those annotations into dependency graphs, which we think is promising for future work. After distribution, our training corpus has a very different qualitative look. There are treelike subgraphs and then densely connected clusters of EDUs, indicating the presence of CDUs. This gives us good reason to believe that in subsequent work, we will be able to predict CDUs and attack the problem of hierarchical discourse structure seriously.

References

Stergos Afantenos, Nicholas Asher, Farah Benamara, Anas Cadilhac, Cdric Degremont, Pascal Denis, Markus Guhe, Simon Keizer, Alex Lascarides, Oliver Lemon, Philippe Muller, Soumya Paul, Verena Rieser, and Laure Vieu. 2012. Developing a corpus of strategic conversation in the settlers of catan. In Noriko Tomuro and Jose Zagal, editors, *Workshop on Games and NLP (GAMNLP-12)*, Kanazawa, Japan.

Stergos Afantenos, Eric Kow, Nicholas Asher, and Jérémy Perret. 2015. Discourse parsing for multiparty chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 928–937, Lisbon, Portugal, September. Association for Computational Linguistics.

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Studies in Natural Language Processing. Cambridge University Press, Cambridge, UK.

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer Academic Publishers.

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

A. Berger, S. Della Pietra, and V. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Chloé Braud and Pascal Denis. 2015. Comparing word representations for implicit discourse relation classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2201–2211, Lisbon, Portugal, September. Association for Computational Linguistics.

Anais Cadilhac, Nicholas Asher, Farah Benamara, and Alex Lascarides. 2013. Grounding strategic conversation: Using negotiation dialogues to predict trades in a win-lose game. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 357–368, Seattle, Washington, USA, October. Association for Computational Linguistics.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In Jan van Kuppevelt and Ronnie Smith, editors, *Current Directions in Discourse and Dialogue*, pages 85–112. Kluwer Academic Publishers.

Dipanjan Das, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56, March.

Pascal Denis and Jason Baldridge. 2007. Joint determination of anaphoricity and coreference resolution using integer programming. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 236–243, Rochester, New York, April. Association for Computational Linguistics.

Pascal Denis and Philippe Muller. 2011. Predicting globally-coherent temporal structures from texts via endpoint inference and graph decomposition. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*.

David duVerle and Helmut Prendinger. 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, August. Association for Computational Linguistics.

Jason M. Eisner. 1996. Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, volume 1, pages 340–345, Copenhagen, Denmark.

Micha Elsner and Eugene Charniak. 2010. Disentangling chat. *Computational Linguistics*, 36(3):389–409.

Micha Elsner and Eugene Charniak. 2011. Disentangling chat with local coherence models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1179–1189, Portland, Oregon, USA, June. Association for Computational Linguistics.

Vanessa Wei Feng and Graeme Hirst. 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, July. Association for Computational Linguistics.

Daniel Fernández-González and André F. T. Martins. 2015. Parsing as reduction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1523–1533, Beijing, China, July. Association for Computational Linguistics.

Hugo Hernault, Helmut Prendinger, David A. duVerle, and Mitsuru Ishizuka. 2010. HILDA: A Discourse Parser Using Support Vector Machine Classification. *Dialogue and Discourse*, 1(3):1–33.

Tsutomu Hirao, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda, and Masaaki Nagata. 2013. Single-document summarization as a tree knapsack problem. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1515–1520, Seattle, Washington, USA, October. Association for Computational Linguistics.

Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad. 2013. Combining intra- and multi-sentential rhetorical parsing for document-level discourse analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 486–496, Sofia, Bulgaria, August. Association for Computational Linguistics.

Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2015. Codra: A novel discriminative framework for rhetorical analysis. *Computational Linguistics*.

Sujian Li, Liang Wang, Ziqiang Cao, and Wenjie Li. 2014. Text-level discourse dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25–35, Baltimore, Maryland, June. Association for Computational Linguistics.

Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. 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, August. Association for Computational Linguistics.

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

William C. Mann and Sandra A. Thompson. 1988. Rhetorical Structure Theory: Towards a Functional Theory of Text Organization. *Text*, 8(3):243–281.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Daniel Marcu and Abdessamad Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *Proceedings of ACL*, pages 368–375.

Andre Martins, Noah Smith, Eric Xing, Pedro Aguiar, and Mario Figueiredo. 2010. Turbo parsers: Dependency parsing by approximate variational inference. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 34–44, Cambridge, MA, October. Association for Computational Linguistics.

Ryan T. McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. 2005. Non-projective dependency parsing using spanning tree algorithms. In *HLT/EMNLP*.

Philippe Muller, Stergos Afantenos, Pascal Denis, and Nicholas Asher. 2012. Constrained decoding for text-level discourse parsing. In *Proceedings of COLING 2012*, pages 1883–1900, Mumbai, India, December. The COLING 2012 Organizing Committee.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie L. Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of LREC 2008*.

Kenji Sagae. 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.

R. Soricut and D. Marcu. 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.

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

Rajen Subba and Barbara Di Eugenio. 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, June. Association for Computational Linguistics.

Maite Taboada and William C. Mann. 2006. Rhetorical Structure Theory: Looking Back and Moving Ahead. *Discourse Studies*, 8(3):423–459, June.

Li Wang, Marco Lui, Su Nam Kim, Joakim Nivre, and Timothy Baldwin. 2011. Predicting thread discourse structure over technical web forums. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 13–25, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.