THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *22 décembre 2016* **par :**

Jérémy PERRET

Parsing dialogue and argumentative structures

JURY

Leila AMGOUD	Directeur de Recherche	Présidente du Jury
Alexis NASR	Professeur	Membre du Jury
Laurent PREVOT	Professeur	Membre du Jury
Ekaterina SHUTOVA	Chercheur	Membre du Jury
Nicholas ASHER	Directeur de Recherche	Membre du Jury
Stergos AFANTENOS	Maître de Conférences	Membre du Jury

École doctorale et spécialité :
MITT : Domaine STIC : Intelligence Artificielle
Unité de Recherche :
Institut de Recherche en Informatique de Toulouse (UMR 5505)
Directeur(s) de Thèse :
Nicholas ASHER et *Stergos AFANTENOS*
Rapporteurs :
Alexis NASR et *Laurent PREVOT*

Abstract

This work presents novel techniques for parsing the structures of multi-party dialogue and argumentative texts. Finding the structure of extended texts and conversations is a critical step towards the extraction of their underlying meaning. The task is notoriously hard, as discourse is a high-level description of language, and multi-party dialogue involves many complex linguistic phenomena.

Historically, representation of discourse moved from local relationships, forming unstructured collections, towards trees, then constrained graphs. Our work uses the latter framework, through Segmented Discourse Representation Theory. We base our research on a annotated corpus of English chats from the board game *The Settlers of Catan*. Per the strategic nature of the conversation and the freedom of online chat, these dialogues exhibit complex discourse units, interwoven threads, among other features which are mostly overlooked by the current parsing literature.

We discuss two corpus-related experiments. The first expands the definition of the Right Frontier Constraint, a formalization of discourse coherence principles, to adapt it to multi-party dialogue. The second demonstrates a data extraction process giving a strategic advantage to an artificial player of *Settlers* by inferring its opponents' assets from chat negotiations.

We propose new methods to parse dialogue, using jointly machine learning, graph algorithms and linear optimization, to produce rich and expressive structures with greater accuracy than previous attempts. We describe our method of constrained discourse parsing, first on trees using the Maximum Spanning Tree algorithm, then on directed acyclic graphs using Integer Linear Programming with a number of original constraints.

We finally apply these methods to argumentative structures, on a corpus of English and German texts, jointly annotated in two discourse representation frameworks and one argumentative. We compare the three annotation layers, and experiment on argumentative parsing, achieving better performance than similar works.

Résumé

Le présent manuscrit présente de nouvelles techniques d'extraction des structures : du dialogue de groupe, d'une part; de textes argumentatifs, d'autre part. Déceler la structure de longs textes et de conversations est une étape cruciale afin de reconstruire leur signification sous-jacente. La difficulté de cette tâche est largement reconnue, sachant que le discours est une description de haut niveau du langage, et que le dialogue de groupe inclut de nombreux phénomènes linguistiques complexes.

Historiquement, la représentation du discours a fortement évolué, partant de relations locales, formant des collections non-structurées, vers des arbres, puis des graphes contraints. Nos travaux utilisent ce dernier paradigme, via la *Théorie de Représentation du Discours Segmenté.*[1] Notre recherche se base sur un corpus annoté de discussions en ligne en anglais, issues du jeu de société *Les Colons de Catane*. De par la nature stratégique des conversations, et la liberté que permet le format électronique des discussions, ces dialogues contiennent des *Unités Discursives Complexes*,[2] des fils de discussion intriqués, parmi d'autres propriétés que la littérature actuelle sur l'analyse du discours ignore en général.

Nous discutons de deux investigations liées à notre corpus. La première étend la définition de la contrainte de la frontière droite,[3] une formalisation de certains principes de cohérence de la structure du discours, pour l'adapter au dialogue de groupe. La seconde fait la démonstration d'un processus d'extraction de données permettant à un joueur artificiel des *Colons* d'obtenir un avantage stratégique en déduisant les possessions de ses adversaires à partir de leurs négociations.

Nous proposons de nouvelles méthodes d'analyse du dialogue, utilisant conjointement apprentissage automatisé, algorithmes de graphes et optimisation linéaire afin de produire des structures riches et expressives, avec une précision supérieure comparée aux efforts existants. Nous décrivons notre méthode d'analyse du discours par contraintes, d'abord sur des arbres en employant la construction d'un arbre couvrant maximal, puis sur des graphes orientés acycliques en utilisant la programmation linéaire par entiers avec une collection de contraintes originales.

Nous appliquons enfin ces méthodes sur les structures de l'argumentation, avec un corpus de textes en anglais et en allemand, parallèlement annotés avec deux structures du discours et une argumentative. Nous comparons les trois couches d'annotation et expérimentons sur l'analyse de l'argumentation, obtenant de meilleurs résultats, relativement à des travaux similaires.

[1] Segmented Discourse Representation Theory.
[2] Complex Discourse Units.
[3] Right Frontier Constraint.

Remerciements

Ce manuscrit n'aurait jamais vu le jour sans le soutien sans faille de nombreuses personnes, à qui je dois toute ma gratitude.

En tout premier lieu, mes encadrants, Nicholas Asher et Stergos Afantenos, que je remercie pour leurs précieux conseils, leur infinie patience et leur franchise ;

Mon jury, Leila Amgoud, Laurent Prévot, Alexis Nasr et Ekaterina Shutova, pour leur lecture et commentaires sur mes travaux ;

L'équipe du projet STAC et l'équipe MELODI de l'IRIT, pour les discussions enrichissantes de ces trois ans.

Sur un plan plus personnel, mes remerciements à Eric Kow, pour ses conseils de productivité et d'attention ; à ma famille pour avoir supporté mon humeur et mon emploi du temps plus que chaotiques ; à mes amis pour leur oreille attentive, leurs encouragements et leur bonne humeur.

Merci, enfin, à vous lecteur, de tolérer le style quelque peu brut de ce manuscrit. N'hésitez pas à me contacter au besoin. Pour la Science.

Contents

Chapter 1

Introduction

This thesis belongs to the domain of Natural Language Processing (NLP), the sub-domain of Artificial Intelligence (AI) dedicated to the task of manipulating human languages, like English or French, as opposed to artificial ones like Python or HTML. The latter have rigorously defined syntax and semantics. Natural language does not, it's ambiguous, full of exceptions, it needs context to be correctly interpreted, and it's constantly evolving. Yet, humans use it fluently, without noticing the feat.

Consider, for instance, the following conversation:

- Alice: I'm hungry.
- Bob: Me too. Who's up for pizza?
- Carol: Pizza again? We just had some yesterday.
- Bob: So what? Everyone loves it. Especially Dan.
- Dan: Hey guys, sorry for being late.
- Alice: Hey Dan, we're about to order pizza.
- Dan: Pizza again? Of course!
- Bob: Told you he would love it.
- Alice: Ordering the usual, pepperoni, hawaiian, cheese?
- Dan: Sounds good.
- Bob: Great.
- Carol: Eh, ok. I'm too hungry to be picky anyways.

As a reader, you probably had no difficulty figuring out the flow of this conversation. You perform a similar task every day, without noticing, and every person you interact with does too.

We, humans, are able to use language as a very rich, powerful and ubiquitous tool to convey meaning, to communicate ideas with others in an ordered manner. We infer, from text or speech, the implicit structure of conversation: who's answering to whom, the topic of the conversation, keeping track of multiple topics

at once, and so on. All these items are necessary to understand the meaning of a conversation, but it also works for any text directed at someone. When reading a newspaper article, the ideas aren't jumbled together: they are organized so that the reader could follow the train of thought of the writer, as if the text was read aloud to them.

Language not only conveys facts and observations, but also commitments to rhetorical stances. Carol, for example, in the conversation above, was initially opposed to pizza, and begrudgingly accepted it at the end, following the stated wishes of everyone else to order some.

Such argumentative stances are fundamental to any debate. The facts have a persuasive role, which we don't always perceive consciously, but that we process anyways, updating our beliefs endlessly and effortlessly.

All of this suggests a natural language text possesses a structure, a non-random underlying organization. The two domains of argumentation and discourse involve a set of content units linked by various semantic relations. One question is whether the units and semantic relations are the same across domains, or share interesting features. Another natural question is whether methods developed in one domain can be transported to the other.

This thesis will make headway in answering these two questions. However, there is much to be done before these questions have a definitive answer. The methods for extracting discourse structures from texts, let alone dialogues, is still in its infancy, and a large part of the thesis details my efforts to further discourse parsing (see chapter 6). We also have investigated methods for extracting argumentation structures from texts (see chapter 7).

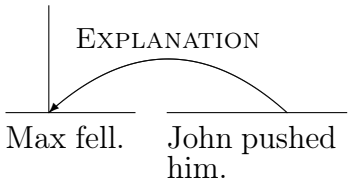
In general our work shows that discourse structures and argumentation structures have important and deep similarities, even if the basic constituents of the structures may very well differ, and even though the relations used in one domain may only be a subset of those used in another.

We will give now an chapter-by-chapter outline of the thesis. We will describe successively the background of our research in discourse and argumentation; then the *Settlers* corpus, the primary dataset for our study of discourse, and investigation thereof; finally our work and experiments in discourse and argumentative parsing, before concluding.

In chapter 2, we will describe past research on the representation of discourse, and the automated extraction of its structure. All representation frameworks rely on the existence of *discourse relations*, linking parts of a text, with associated semantics and behavior. Consider the following example:

(1.1) Max fell. John pushed him.

The first sentence describes an event, the second its cause; we describe the discourse relation linking the two semantically by giving it a label (here EXPLANATION) and linking the two:



We describe three discourse representation frameworks:

- *Rhetorical Structure Theory* (henceforth RST), developed by (Mann and Thompson, 1987; Mann and Thompson, 1988; Taboada and Mann, 2006), describes a hierarchical tree-based global structure of texts;
- *Segmented Discourse Representation Theory* (SDRT), developed by Asher (1993), describes a global structure of texts as well, based on hypergraphs;
- The *Penn Discourse TreeBank* (PDTB) framework, developed by (Miltsakaki et al., 2004; Prasad et al., 2008), describes local relations only between pairs of text spans.

While each of them have their own section in chapter 2, we will examine them jointly in this introduction.

To describe the structure of a whole text, me must first describe the entities which will be linked together, by splitting the text in *Elementary Discourse Units* (henceforth EDUs); we call this process *segmentation*. All frameworks possess discourse units, although their formal definition and semantics differ. Consider the following segmentation of a news excerpt:

(1.2) [Interprovincial Pipe Line Co. said]$_1$ [it will delay a proposed two-step, 830 million dollar expansion of its system]$_2$ [(US$705.6 million)]$_3$ [because Canada's output of crude oil is shrinking.]$_4$

In the hierarchical view of RST, a text (considered as a root span) is split in *children spans* linked together by a rhetorical relation. Each child is then split recursively, until a span cannot be broken down by a relation. The resulting atomic spans, contiguous and mutually exclusive, represent the EDUs of the text. A given span always being contiguous, this also means all of its children must remain adjacent: as a result, RST trees are projective (with no crossing dependencies).

The rhetorical relations are defined by a *label* and a *nuclearity*. The label, such as EXPLANATION, CONTRAST, SUMMARY or BACKGROUND, represent the

semantic roles played by the children. Those are clearly defined by a natural language description, while having no formal interpretation. The nuclearity determines which children span is the *Nucleus* of the relation, of which the other spans are the subordinate *Satellites*. Most relations accept two arguments only (one Nucleus and a Satellite) while some rare *multinuclear* relations such as LIST have multiple nuclei, all of the same importance.

For example, the EVIDENCE relation is used when the author thinks the Satellite increases the belief of the reader in the Nucleus.[1]

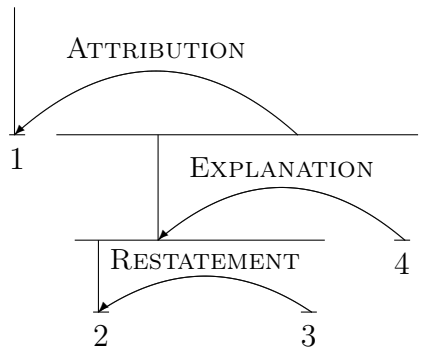


Figure 1.1: RST representation for example 1.2, with arrows pointing to the Nucleus.

A key element to the interpretation (and construction) of RST structures is the *Nuclearity Principle*: a rhetorical relation between two spans should hold between the *recursive Nuclei* of the spans, which is a restrictive model (see next paragraph), but allows quick summarization of spans by reducing them to their Nucleus.

SDRT has its roots in Kamp's Discourse Representation Theory (Kamp et al., 2011), aiming to produce a logical interpretation of discourse structure through the combination of the logical form of its components. Thus, EDUs in SDRT are text spans corresponding to the atomic clauses of the text. A SDRT structure can be described as a directed graph between two kinds of units: EDUs, and *Complex Discourse Units* (CDUs). CDUs are clusters of DUs acting as arguments for discourse relations.

In figure 1.2, the CDU π contains both EDUs 2 and 4 but not 3, meaning the target of the ATTRIBUTION relation, i.e. *what Interprovincial Pipe Line Co. said* leaves out the currency reformulation. This enables SDRT to have finer scoping rules than RST. SDRT also allows crossing dependencies: there are no limitations

[1]Which means an RST annotator should interpret the author's intention when identifying the relations. This is intended.

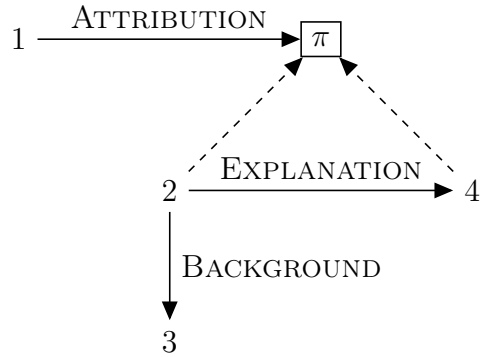


Figure 1.2: SDRT representation of example 1.2. Arrows mark discourse relations, dashed lines mark inclusion in a CDU.

in the formal definition regarding adjacency of arguments of relations, allowing crossing dependencies and even units being the arguments of multiple relations, e.g. EDU 2 in example 1.2.

There are, however, formal constraints regarding units able to be linked. The first one is *acyclicity*: a SDRT relation represents a reference to an earlier context, which is very often *anaphoric* (a unit referring to an earlier unit), but sometimes *cataphoric* (the opposite). In any case, references cannot be circular.

The second constraint, stemming from observation of the coherence of discourse, is the *Right Frontier Constraint*. The concept of right frontier denotes the current context of interpretation of discourse by a reader (or listener). Compare the following examples:

(1.3) Rose dumped the cookies on the floor.$_{d_1}$ (So) She was sent to her room.$_{d_2}$ (And) She drew all over the kitchen wall.$_{d_3}$

(1.4) Rose dumped the cookies on the floor.$_{e_1}$ (And) She drew all over the kitchen wall.$_{e_2}$ (So) She was sent to her room.$_{e_3}$[2]

The first example is intuitively incoherent: the fact that Rose drew over the wall no longer seems *relevant* to the context, as the consequence (sent to her room) was already explored. The two examples are represented in the following way:

RESULT is a *coordinating* relation, which "closes access" to its first argument for all later EDUs. In contrast, ELABORATION is a *subordinating* relation, which still enables access to its first argument. In example 1.4, a RESULT relation may well join EDUs e_1 and e_3 without incoherence. We separate SDRT relations between subordinating and coordinating.

[2]A discourse marker or two would render the discourse less choppy, though markers are not needed to achieve the intended interpretation.

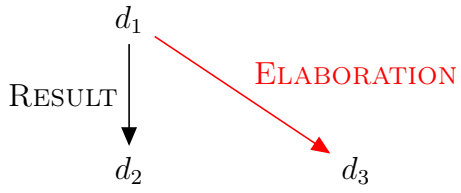


Figure 1.3: Example 1.3

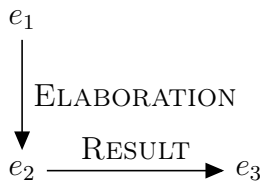


Figure 1.4: Example 1.4

The *Right Frontier Constraint* (RFC) is applied to any new EDU being added to a discourse graph, filtering which existing "accessible" EDUs it can attach to. This creates a "right frontier", due to the position of the accessible EDUs when drawing the SDRT graph; hence the name.

While this constraint isn't absolute in SDRT's definition, we hypothesize it matches closely the behavior of natural conversation. We test this assertion empirically on the *Settlers* corpus. The formal definition, accuracy and filtering power of the RFC are discussed in section 2.3.

The PDTB is a *dataset*, created specifically for the study of discourse. Its annotation model aims to be *theory-neutral*, by describing informally a wide number of relations. The primary focus of the PDTB are *discourse connectives* linking two (and always two) spans of text; importantly, the authors of the corpus do not ambition to describe the whole structure of text, but only local relations.

In example 1.2, spans 2 and 4 are linked by the connective *because* with a CONTINGENCY.CAUSE.REASON relation. The connective is here *explicit* as it appears in the text, but connectives may also be *implicit*. Compare the following:

(1.5) Max fell, because John pushed him.

(1.6) Max fell. John pushed him.

The relation is the same in both examples, however without connective in the second. In the PDTB, two adjacent spans without a connective are annotated with the implicit connective that matches best their relation.[3] To help disambiguate polysemous connectives (e.g. *since*, which can have a temporal and/or causal interpretation), PDTB offers a detailed sense annotation hierarchy (shown in section 2.1.2). Reusing the previous example, the REASON sense belongs to the category CAUSE, itself belonging to the top-level category CONTINGENCY.

Few large-scale annotated datasets are dedicated to discourse structure. PDTB itself is one, albeit for local discourse relations only. We also describe the RST Discourse Treebank (composed of news articles annotated in RST fashion) and the

[3]In case they are completely unrelated, there is a connective for that too: *NoRel*.

Settlers corpus (online game chats annotated in SDRT fashion), the latter being extensively described in chapter 4.

In section 2.2, we move on to discourse parsing: the task of extracting all of the previously described structures automatically. As a description of the high-level organization of text, they can be used in a variety of applications, such as text summarization, coreference resolution, data mining, conversational interfaces, among many others. We detail a direct application of parsing in section 5.2.

The history of discourse parsing follows closely the development of new representation frameworks and, more importantly, of reliable annotated datasets to experiment on.

Focusing on the analysis of local relations, Marcu (2000) relies on discourse markers (akin to explicit PDTB connectives) to build a brittle parser of RST structures. Faced with the high dimensionality of semantic space and the combinatorial nature of vocabulary,[4] the discourse parsing community quickly relied on machine learning methods to classify discourse relations.

However, reliable data is scarce. Marcu and Echihabi (2002) also use discourse markers to automate annotations on a large dataset, to predict implicit relations from lexical cues. Sporleder and Lascarides (2005) use a similar approach to disambiguate between a restricted set of relations.

The publication of the Penn Discourse TreeBank drove a fair amount of research on local structure. Efforts in the domain are mostly incremental, and rely on shallow features for detection or classification of relations (Wellner and Pustejovsky, 2007; Pitler et al., 2009; Lin et al., 2009; Zhou et al., 2010). Frequently used are the syntactic structure of texts (syntactic heads in particular), organisational features (where is the connective placed in the sentence, how long is the sentence, etc.), and lexical features (word count, keywords, lexical patterns). More recent work uses original features from distributional semantics, such as Brown clusters (Rutherford and Xue, 2014) and word embeddings (Braud and Denis, 2015).

In parallel, the extraction of full discourse structures receive fewer attention. The vast majority of works analyze the RST Discourse Treebank,[5] thus parsing projective trees. The majority of the methods use bottom-up parsing, using greedy algorithms (Soricut and Marcu, 2003; duVerle and Prendinger, 2009; Hernault et al., 2010), shift-reduce parsing (Sagae, 2009; Subba and Di Eugenio, 2009) or CKY parsing (Joty et al., 2012). Works are also split whether they restricted themselves to the sentence scope (which is easier), or attempted to parse full texts.

[4] In other words, language being *vastly* too powerful and expressive to be bound by handmade rules.

[5] Unsurprisingly, as reliable data is scarce and it was for a long time the largest corpus of the domain by far.

With one notable exception (Baldridge and Lascarides, 2005), all the aforementioned work focus on monologue. We base our own research on the first large-scale corpus of annotated dialogue, exploring the limits of the recent monologue-centric methods and expanding the scope of dialogue parsing to new horizons.

In chapter 3, we will describe the past research in the representation of argumentation, and the automated extraction of its structure. We won't give a detailed account of the very abundant literature on the domain, as we focus on the construction of semi-formal argumentation structure, which require relatively basic concepts, and avoid lengthy discussions of the linguistic nature of persuasion.

Classical study of argumentation are based on the validity of arguments, expressed through logic, and the various rhetorical means to *persuade* people. The work of Toulmin (1958), focusing on the *practical* way arguments are organized in language has been extremely influential; based on legal arguments, he separates the statements of a persuasive text by function: the core claim and conclusion of the text; the facts and evidence to back it; the reasoning to extract the conclusion from the facts; the credentials to support that reasoning; finally qualifiers of the strength of the final assertion, and possible exceptions to the reasoning.

Another influential approach is Van Eemeren and Grootendorst (1992), initiating *pragma-dialectical* theory. They emphasize the use of *standpoints*, i.e. stances taken towards particular claims. Parties in a debate advance new standpoints and attack others, according to certain *rules for a critical discussion* that must be observed to avoid fallacious reasoning. The pragma-dialectical view treats argumentation as a complex speech act, intimately linked with discourse.

The logic-based view of Dung (1995) explores a simplified and formal version of argumentative structure, reducing it to a directed graph of abstract claims attacking each other. Dung expresses the interactions between sets of arguments in great detail, influential to the automated processing of idealized negotiations between artificial agents.

We use in our work the framework of Freeman (2011), synthesized by Peldszus and Stede (2013). An argumentative text is there split into *argumentative discourse units* (ADUs), which are sorted by their stance towards the core claim of the text: proponent, or opponent. ADUs are then organized in a tree structure; claims that directly support or attack another claim are linked together in the graph. The framework also supports the notion of *undercutting*, where the inference itself, holding between a supporting claim and its target, is attacked:

- A: Carthage is threatening Rome.
- A: Therefore, Carthage must be destroyed.
- B: Maybe we could negotiate with the threat instead.

We will then review the domain of argumentative parsing, which has been focused mainly on the detection of argumentative claims and stances in text (Moens et al., 2007; Palau and Moens, 2009; Florou et al., 2013) rather than their interactions. The prediction of complete argumentative structures is extremely recent and employ statistical models to classify stances and relations (Peldszus and Stede, 2015; Persing and Ng, 2016; Stab and Gurevych, 2016).

In chapter 4, we will describe in detail the *Settlers* corpus, on which our work in chapters 5 and 6 is based. The corpus comprises annotated text chats from an online version of the board game *The Settlers of Catan*. The game itself involves a group of players competing for resources, trading and negotiating, the bargaining being expressed through free-form text-based discussion. Dozens of such games have been annotated with the SDRT framework. The corpus exhibits many interesting features specific to dialogue, crossing dependencies, long-distance anaphoric links, complex discourse units, interwoven threads of discussion, abbreviated language, among others.

234	gwfs	anyone got wheat for a sheep?
235	inca	sorry, not me
236	Ccg	nope. you seem to have lots of sheep!
237	gwfs	yup baaa
238	dmm	i think i'd rather hang on to my wheat i'm afraid
239	gwfs	kk I'll take my chances then...

Figure 1.5: Excerpt from the *Settlers* corpus.

Table 1.5 presents an excerpt from the corpus, a failed negotiation between four players. Negotiation for resources typically happen once every player turn, starting a conversation. Sometimes, bargaining session are continued over several player turns. The games are thus split in *dialogues*, which can mostly be taken in isolation context-wise. Those are further split in *dialogue turns*, comprised of the utterances of single players (consecutive statements being grouped in the same turn); which are further split into *elementary discourse units* (EDUs), the atomic elements of discourse representation.

The games have been annotated with the SDRT framework, adding discourse relations between units, as well as *complex discourse units*, clusters of DUs acting as higher-level arguments for relations.

Table 1.6 summarizes the main statistics of the *Settlers* corpus. As mentioned earlier, it is the largest corpus of annotated dialogue structure at the time of this writing, providing a solid base for future study.

Dialogues	1081
Turns	9160
EDUs	10678
CDUs	1284
Relation instances	10513

Figure 1.6: Statistics for the *Settlers* corpus.

In chapter 5, we describe two works exploiting the corpus for very different purposes. The first work (Hunter et al., 2015) concerns the *right frontier constraint*, which has been mentioned previously, and its adaptation to multi-party chat. Our observation of interwoven threads of conversation makes the original definition of the RFC brittle, as multiple contexts are defined at once by each participant. We thus propose, and test, a new definition of the RFC for multi-party chat, which is mechanically less restrictive (more units become accessible as a player has more threads to answer to), still retaining a high filtering power.

The second work (Perret et al., 2014) develops a practical application of discourse parsing to data extraction. Our pipeline identifies, from the dialogues from the *Settlers* corpus, the resources revealed by the players during their negotiations. Knowing the possessions of opponents during the game leads to strategic advantages for the player, who can propose more efficient trades. Our system uses a maximum entropy model to classify the dialogue turns revealing resources, another one to detect question-answer pairs (and potential anaphora), and finally a set of rules to extract the type and quantity of the revealed resources.

In chapter 6, we present our efforts on discourse parsing, using the *Settlers* corpus. We attempt to build the structure of dialogues, which are already segmented. We have followed two consecutive approaches on the task.

In a first time, we (Afantenos et al., 2015) expand on the work of Muller et al. (2012b), who focused on SDRT-annotated monologue. We employ a probabilistic local model of discourse relations to pairs of EDUs, trained with the Maximum Entropy method. We used shallow features for training, mostly positional and lexical features, as well as syntactic parsing and dialogue act parsing. The local model is then used as input to a *decoding* process, which optimizes the global discourse structure generated, through the Maximum Spanning Tree algorithm, as opposed to the classic parsing methods based on series of local decisions.

Our parsing model creates and trains on dependency structures. While those are isomorphic to an EDU-only graph with labeled relations, the SDRT framework, and the *Settlers* corpus as a result, contains complex discourse units. We eliminate them using a *head replacement strategy*, described in greater detail in the chapter.

In a second time, as we showed that trees aren't expressive enough to describe multi-party chat discourse structures, we (Perret et al., 2016) experiment with Integer Linear Programming to build directed acyclic graphs from an enhanced local model. The method also optimizes the resulting structure globally, and enables the creation of original constraints fine-tuned for dialogue. We describe and justify each of the constraints used for decoding.

Additionally, we present two novel ways to convert CDUs to obtain dependency graphs, to match more closely the semantics of relations involving clusters of EDUs. This creates three versions of our corpus, on which we evaluate our methods.

In chapter 7, drawing from Stede et al. (2016), we describe the construction of a corpus of argumentative texts, expanded from Peldszus and Stede (2016) to feature three layers of annotations: RST and SDRT for discourse structure, and the framework described in Peldszus and Stede (2013) for argumentative structure. After comparing the three layers, we apply Integer Linear Programming methods to the task of extracting argumentative structures, with a dedicated local model.

In chapter 8, we give a summary of our main contributions and project the continuation of our work.

Chapter 2

Background: discourse and dialogue

2.1 Discourse representation theories

The introduction outlined how we can define discourse relations between pairs of units. This segues into defining a discourse structure for a whole text. While the structure of syntax is clearly defined as a tree, and has very formal constraints regarding the nature of its components, discourse units are much more loosely defined. In this section, we review the various formalisms to represent discourse, exploring the following questions: does discourse have a unambiguous structure? Can we define properly discourse relations? How many kinds of them are there? Can we interpret them?

For instance, Hobbs (1985) describes the various coherence relationships binding spans of text together. From his paper:

(2.1) A: John can open Bill's safe.
A: He knows the combination.

In this example, both utterances are linked by an ELABORATION relation, as the second sentence expands the information expressed in the first. Hobbs describes the semantics of multiple kinds of relations happen throughout text. No formal interpretation is given here.

2.1.1 Rhetorical Structure Theory

Rhetorical Structure Theory (RST), developed by Mann and Thompson (1987), expanded by Mann and Thompson (1988) and Taboada and Mann (2006), formalizes the segmentation of text and defines a set of relations, with defined structural behavior, creating a full treelike discourse structure.

When defining a RST structure, the text is is split in contiguous atomic spans. Every span is then assigned a parent span, which encloses it along with other adjacent spans, and so on recursively. The only exception is the root span, which has no parent and encloses the entire text. Sibling spans (having the same parent) can be linked together in the following fashions:

- Nucleus-Satellite: one sibling, the nucleus, is the principal component of the asymmetric relation it has with the other siblings, the satellites. The nucleus usually contains the core claim of the parent span. The satellites are typically subordinate clauses, contain optional information.

- Multinuclear: no sibling is particularly salient; instances are contrast relations, sequence relations, where sibling spans depend on each other to carry the meaning of the parent span.

The resulting structure is a projective tree, as sibling spans must always be adjacent.

Consider the following example, slightly different from the introduction, as the restatement is embedded in the middle of the span:

(2.2) [Interprovincial Pipe Line Co. said]$_{a1}$ [it will delay a proposed two-step, 830 million dollar]$_{a2}$ [(US\$705.6 million)]$_{a3}$ [expansion of its system]$_{a4}$ [because Canada's output of crude oil is shrinking.]$_{a5}$

The corresponding RST structure in figure 2.1 contains several features. There are directed arcs labeled with discourse relations like EXPLANATION or ATTRIBUTION, where the target of the directed arc is designated as a nucleus, while other components are designated as satellites of the relations. There are also unlabeled, horizontal lines that pick out the spans potentially related by the relations, and vertical lines that link spans to sub-spans. The spans themselves may consist of one or more discourse units or distinguished sub-spans.

More formally, RST trees are typically understood in computational terms as binary trees. But as RST annotations countenance relations that may have an arity greater than 2, we give a general definition, isolating out binary trees as a special case.

While such structures are familiar to most researchers on discourse structure, a rigorous interpretation of it was never part of RST, and is rarely discussed in computational linguistics. The first thing to notice is a possible ambiguity in what might be the terms of a discourse relation; for example, the Attribution relation might hold between the discourse constituent/span on the left *Interprovincial Pipe Line Co. said* and the span consisting of the following three segmented units or some subset of these.

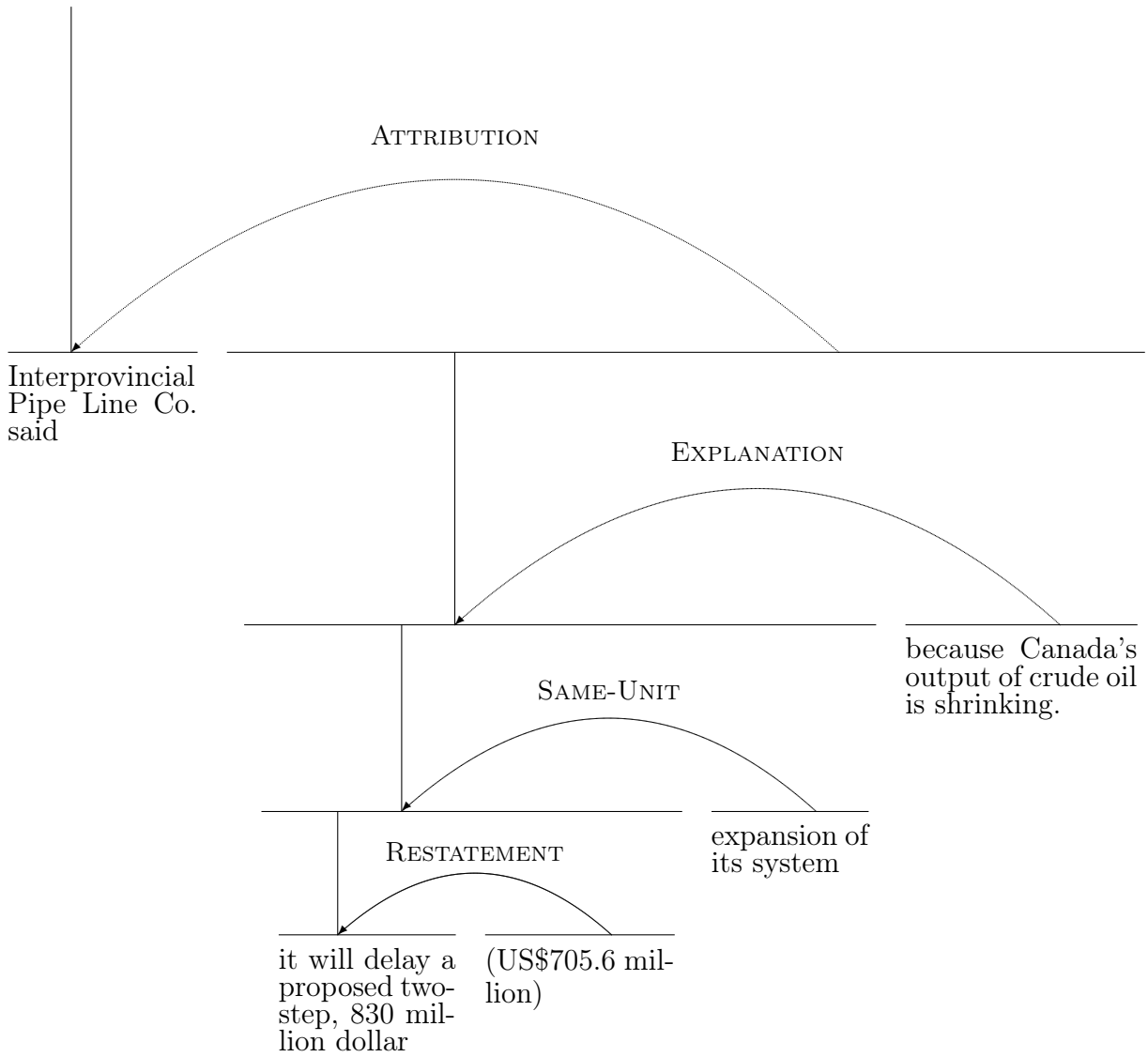


Figure 2.1: RST representation, with arrows pointing to the Nucleus.

In the example at hand, it is obvious from the context that Interprovincial Pipe Line Co. said that it will delay the expansion of its system, and it's also quite probable that what they said didn't include the content in which 830 Canadian dollars are specified in U.S. dollar amounts. Concerning the last discourse unit *because Canada's output of crude oil is shrinking*, it's unclear whether this was part of what Interprovincial Pipe Line Co. said or not.

In RST, we can represent this ambiguity by other making the right term of the ATTRIBUTION relation the value of iteratively seeking the nucleus of a span until one comes to a basic span that has no discourse structure beneath it. We'll call such spans *Elementary Discourse Units*. In our example, this idea, which is formalized under the heading of the *Nuclearity Principle*, would net us only the unit *it will delay a proposed two-step, 830 million dollar*. On the other hand, one might choose not to use the Nuclearity Principle and accordingly take the entire span to the right as the argument of the ATTRIBUTION relation. Interestingly, there does not seem to be a mechanism in the RST literature that would yield, as the second argument of the ATTRIBUTION relation, the content given by elementary discourse units number 2, 4 and 5.

Additionally, units 2 and 4 correspond to the same rhetorical unit, split by the embedded unit *(US$705.6 million)*. As spans must remain contiguous, this translates into the structural relation SAME-UNIT, which has no semantic value whatsoever.

Mann and Thompson describe a number of rhetorical relations, each associated with a rough description of the intended behavior of the relation's arguments. For instance, the asymmetric EVIDENCE relation is described as follows:

- **Constraints on Nucleus:** Reader R might not believe N to a degree satisfactory to Writer;
- **Constraints on Satellite:** Reader believes Satellite or will find it credible;
- **Constraints on the combination:** Reader's comprehending Satellite will increase Reader's belief of Nucleus;
- **Effect:** Reader's belief of Nucleus is increased;
- **Locus of the effect:** Nucleus.

Corpora for discourse structure limited themselves to handcrafted illustrative examples. However, there was a growing need for larger datasets, in order to study discourse parsing. This led to the creation of the first large-scale annotation corpus for discourse structure, the RST Discourse Treebank (RST-DT) (Carlson et al., 2003). 385 Wall Street Journal articles, along with their RST structure, annotated by hand. 78 discourse relations are used, partitioned in 16 classes

displayed in table 2.1. Totalizing 21,789 EDUs, the corpus 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), which will be reviewed in section 2.2.

ATTRIBUTION	BACKGROUND	CAUSE	COMPARISON
CONDITION	CONTRAST	ELABORATION	ENABLEMENT
EVALUATION	EXPLANATION	JOINT	MANNER-MEANS
TOPIC-COMMENT	SUMMARY	TEMPORAL	TOPIC-CHANGE

Table 2.1: RST relation classes used in Carlson et al. (2003).

2.1.2 Penn Discourse TreeBank

The *Penn Discourse Treebank* (PDTB) (Miltsakaki et al., 2004; Prasad et al., 2008) is a *dataset*, in opposition to the frameworks of RST and SDRT discussed in this section. However, the corpus has been designed to be *theory-neutral*, while using a particular discourse framework for its underlying annotations[1].

The aim of PDTB is to provide a dataset in which discourse connectives are annotated. This sets apart PDTB from the RST Discourse Treebank for two main reasons: first, PDTB has no objective of describing the full structure of texts, but only the local level of discourse structure represented by connectives. Secondly, the annotated discourse connectives have to be lexically grounded, while RST annotation aims to reflect the interpretation of the structure of a text by a reader. An overall goal is to make the PDTB annotations reliable and unambiguous.

PDTB connectives are central to the dataset. They are divided in two categories.

Explicit connectives are the expressions that signal a discourse relation between parts of a text. They are split into four syntactic classes:

- Subordinating conjunctions: because, although, when, if, as, etc.
- Coordinating conjunctions: and, but, so, nor, or (and paired versions of the latter — neither/nor, either/or)
- Prepositional phrases: as a result, in comparison, on the one hand/on the other hand, etc.
- Adverbs: then, however, instead, yet, likewise, subsequently, etc.

[1]The neutrality of the annotation schema itself is thus debatable.

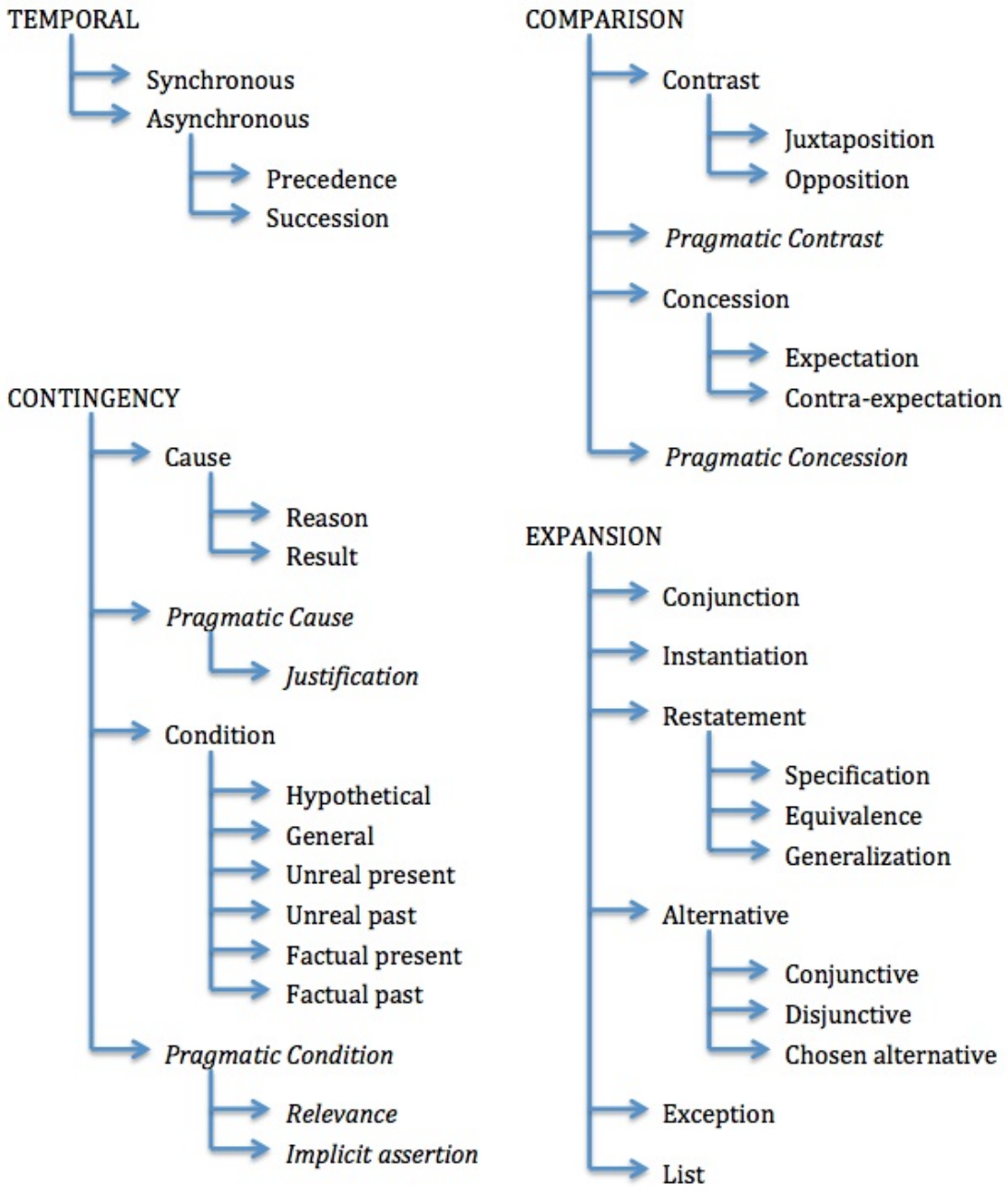


Figure 2.2: PDTB sense hierarchy

Category	Relation count
Explicit	18,459
Implicit	16,224
AltLex	624
EntRel	5,210
NoRel	254
Total	40,600

Table 2.2: Total number of relations annotated in the PDTB, by category.

Implicit connectives "join" two adjacent spans of text where no explicit connective is present. The concept is best illustrated by an example:

(2.3) [Max fell]$_{a1}$, [because John pushed him]$_{a2}$.

(2.4) [Max fell]$_{b1}$. [John pushed him]$_{b2}$.

In the second case, the causal relation is implicit; in PDTB, the pair would be annotated by *IMPLICIT-because*, as the connective matches best the implied relation.

Several additions were made to the PDTB framework following its introduction in 2004, which were eventually gathered into PDTB 2.0 (Prasad et al., 2008), with an actualized annotation manual. Among the new features were three new connective categories, for special cases where an implicit connective couldn't be provided:

- *AltLex*, when the discourse relation is marked by a non-connective expression (such as "One potential cause may be...");
- *EntRel*, when the spans are linked only by an entity-based coherence relation;
- *NoRel*, when no relation at all could be detected between adjacent spans.

Another new feature was refined *sense annotations* for connectives, semantic categories provided for disambiguation. For instance, the connective *since* can have a causal (2.5) or a temporal sense (2.6):

(2.5) Arthur was happy, *since* the cake tasted good.

(2.6) Arthur was happy, *since* the arrival of his guests.

The set of sense tags is organized hierarchically, with four semantic classes at the top: Temporal, Contingency (for causal and conditional connectives), Comparison (for contrast and concession) and Expansion (for conjunction, instantiation, restatement, alternatives, exceptions and lists).

2.1.3 Segmented Discourse Representation Theory

Origins of SDRT As described in the introduction, one of the original goals of discourse representation was to accurately describe, and formalize, the *meaning* of text in logical form. Montague semantics (Montague et al., 1976) aimed to translate the syntactic structure of sentences from the semantic value of their components (ultimately, words). The next step of the bottom-up approach was to move on to multi-sentence texts, and eventually dialogue. In contrast, RST follows a top-down approach to discourse structure, highly hierarchical. Building a correctly labeled structure in RST involves having access to the whole text from the start of the parsing process.

Expanding on Montague's work, Kamp developed *Discourse Representation Theory* (DRT, Kamp (1988)), using an incremental approach to the interpretation problem. The parsing process first builds a logical interpretation of the first sentence. Following sentences are then parsed as additions to the existing context, referring to the previous elements of the text. Any sentence after the first is never viewed as standalone. This approach permits a greater flexibility for discourse structure. An excellent thing, since further study of dialogue proved that the RST constraints weren't expressive enough.

Asher (1986) extended DRT to take account of propositional type discourse entities. Such entities describe the mental state of an agent towards a proposition (belief, fear, hope, etc.) Those were frequently introduced by elements like *that*-clauses, as in

(2.7) Yoda believes that Palpatine is evil.

Asher's analysis of this was roughly:

$$\exists p \quad believes(Yoda, p) \wedge p \approx evil(Palpatine)$$

where $a \approx b$ is defined as the content of a is at least partially specified by b.

Asher noted however that discourse made reference to such abstract entities as propositions even when they were not marked syntactically, e.g. as denotations of *that* clauses. As in:

(2.8) Palpatine is a traitor and a murderer but most Jedi sadly don't realize it.

The *it* picks up the first clause, but in standard semantic theories, including DRT, such entities would not be introduced as variables in any way. Furthermore, Asher noticed that not all proposition level contents could be so picked up. For example,

(2.9) Three students got in trouble. One had copied during an exam; the second had plagiarized someone else's work; the third had bullied other students into doing his homework for him. The teacher found this reprehensible.

Asher observed that the anaphoric antecedents for *this* were quite limited. It could be what all three of the students had done (collectively) or it could be what the last student had done, but it could not pick up what the first or second students had done. To solve this problem Asher (1993) developed *Segmented Discourse Representation Theory* (SDRT), in which all clauses introduced discourse entities. To limit the set of potential antecedents, he then observed that such discourse entities stood in particular semantic relations to each other that governed anaphoric accessibility of propositional discourse referents. This relational structure then allowed him to define a right frontier constraint that served to restrict anaphoric availability.

If SDRT was developed as a theory of abstract entity anaphora, it soon became apparent that the discourse structures (SDRSs) it posited had other uses in semantics. Hobbs (1979) had already observed that discourse relations could affect the temporal structures in texts, and Lascarides and Asher (1993) developed and formalized Hobbs's insights. They also provided the first logical reconstruction of the reasoning required to construct SDRSs from information contained in the clauses that provide the basic discourse units in those representations using a non-monotonic logic developed in Asher and Morreau (1991). Since then SDRT has been applied to analyses of many semantic phenomena: verb phrase ellipsis (Asher et al., 1997), presupposition (Asher and Lascarides, 1998) and many other phenomena both in text semantics and the semantics of dialogue.

Also while SDRT was originally designed to remedy defects of Kamp's DRT, it also soon became apparent that while dynamic semantics was essential to SDRT's analyses of anaphora and temporal structure, the relational structure or discourse structure posited by the theory was compatible with pretty much any dynamic semantics (e.g. Groenendijk and Stokhof (1991) or continuation style semantics (e.g. Asher and Pogodalla (2010)).

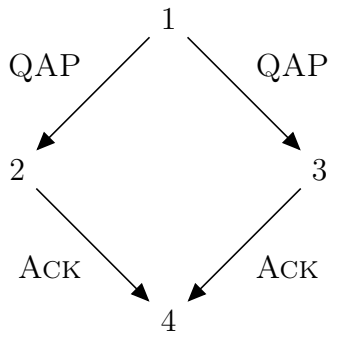
Motivations for a flexible structure While projective trees are arguably a contender for representing the discourse structure of monologue text, they rule out by definition any kind of crossing dependencies. We argue this cripples the expressivity of the framework. In a long monologue, an author might spend a sentence or even some paragraphs to give more details on a topic they mentioned earlier, before resuming their core narrative of sequence. At any other point in the text, the other might delve into a minor subject, referring to entity or events from any point in the previous text, as long as the author remains semantically

coherent (which is a rather weak constraint). The only limit as to how far back the author can make connections is the cognitive ability of the reader.

This kind of crossing dependency appears more frequently in multi-party dialogue. 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, or refer to an observation made much earlier in the discussion. In the case of chat dialogue, the cognitive load required from the participants is drastically reduced, as they all have direct access to the history of the conversation, enabling them to refer to any previous comment easily and often implicitly.

Such freedom rules out using a theory like RST as a basis either for an annotation model or as a guide to learning discourse structure in a more general context. 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 discourse either, as Asher and Lascarides (2003) argue on theoretical grounds. We give more details on the matter in section 4.2.1. As an example, the following dialogue exhibits non-treelike structure:

1. Alice: Is pizza OK for you two?
2. Bob: Yup!
3. Carol: No objection.
4. Alice: Perfect.



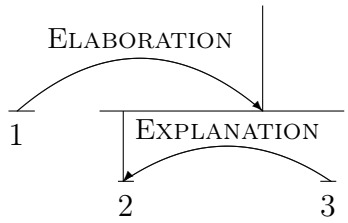
Here, the simultaneous ACKNOWLEDGEMENT from Alice of the two answers of Bob and Carol create an intuitive, "lozenge"-like structure which projective trees cannot represent.

The above observations lead to the use of graphs as discourse structure, which in turn isn't expressive enough for discourse. 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. Consider the following examples, from Asher et al. (2011):

(2.10) [For the last two decades,]$_{a1}$ [the German central bank had a restrictive monetary policy,]$_{a2}$ [because it viewed inflation as the number one problem.]$_{a3}$

(2.11) [John worked at U.T. for two decades $]_{b1}$ [He worked in the library$]_{b2}$ [because he wanted to be in charge of large collections.$]_{b3}$

Here, the RST representation of both sentences is the same:



However, the semantic interpretation of the structure differs. In example 2.10, ELABORATION has scope over both other EDUs: the German viewed inflation as a concern for two decades. In contrast, in example 2.11, ELABORATION doesn't have scope over EDU 3: John didn't want to be in charge of collections for two decades. Using clusters of nodes for representation enables the following distinction:

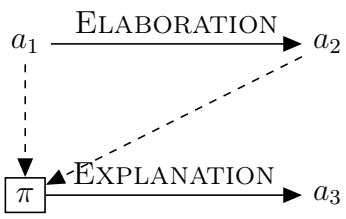


Figure 2.3: Example 2.10

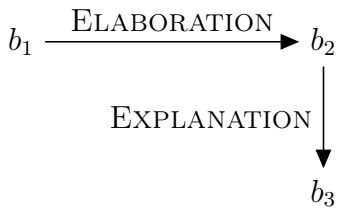


Figure 2.4: Example 2.11

Sub-graphs of the entire discourse graph can thus act as elements or nodes in the full discourse structure. These sub-graphs are called *complex discourse units* or CDUs. Asher (1993) argue they are an important organizing principle of discourse. As we saw in the examples, CDUs enable precise and unambiguous scoping of discourse relations, which is critical to the accurate interpretation of anaphora and ellipsis.

However, although CDUs are present in discourse corpora, especially SDRT-annotated ones, as they are a fundamental component of the framework, very few works have attempted to predict them, substituting them by other structure whenever they appear; we describe these workarounds, and our own propositions on the topic, in sections 6.1.1 and 6.2.1.

SDRT structures We move now to a complete definition. In SDRT, a discourse structure, or SDRS (for *Segmented Discourse Representation Schema*), consists of a set of Discourse Units (DUs) and of discourse relations linking those units. DUs are distinguished into EDUs and CDUs:

– EDUs (elementary discourse units) correspond to phrases or sentences describing a state or an event, the atomic clauses of the text, ideally interpretable as logic predicates;

– CDUs (complex discourse units) are sets of DUs acting as arguments for discourse relations, used whenever a *group* of units act as a single semantic unit.

Formally, for a given text segmented in a set D of EDUs, where $D = \{e_1, \ldots, e_n\}$, an SDRS is a tuple (V, E_1, E_2, ℓ) where:

- $V = D \cup \Pi$ is a set of nodes or discourse units, with Π as the set of CDUs ;
- $E_1 \subseteq V \times V$ is a set of edges representing discourse relations;
- $E_2 \subseteq V \times \Pi$ is a set of edges that represents parthood in the sense that if $(x, y) \in E_2$, then the unit x is a component of the CDU y;
- $\ell \colon E_1 \to$ *Relations* is a labeling function that assigns an edge in E_1 its discourse relation type.

Consider the following example:

(2.12) [Interprovincial Pipe Line Co. said]$_{e1}$ [it will delay a proposed two-step, 830 million dollar [(US$705.6 million)]$_{e2}$ expansion of its system]$_{e3}$ [because Canada's output of crude oil is shrinking.]$_{e4}$

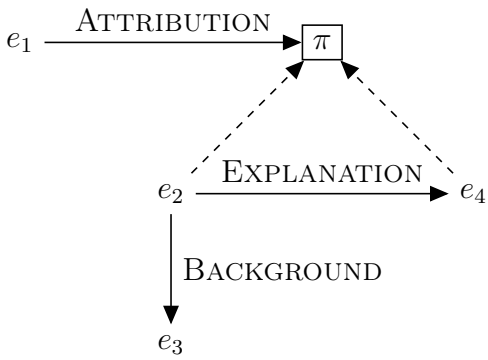


Figure 2.5: SDRT representation of example 2.12.

Here $D = \{e_1, e_2, e_3, e_4\}$, $\Pi = \{\pi\}$, $E_1 = \{(e_1, \pi), (e_2, e_3), (e_2, e_4)\}$ and $E_2 = \{(e_2, \pi), (e_4, \pi)\}$.

While SDRT units cannot partially overlap, inclusion is allowed, e.g. EDUs 2 and 3. The semantic interpretation strips away embedded span, so that the portion of text corresponding to EDU 3 is *it will delay a proposed two-step, 830 million dollar expansion of its system.*

In the corresponding SDRT representation of the example, in figure 2.5, plain arrows correspond to discourse relations (edges in E_1) and dashed arrows to CDU membership (edges in E_2).

SDRT relations We describe informally the set of SDRT relations used in the annotation of the *Settlers* corpus, described in chapter 4.

ELABORATION	β provides extra information about the eventuality described in α
EXPLANATION	β explains why, or gives the cause of, what happened in α
ACKNOWLEDGEMENT	β signals acknowledgement or acceptance of the content of α
Q-ANSWER PAIR	β is the answer to the question α
Q-ELAB[2]	β is a follow-up question to α, requesting more information
CLARIFICATION Q.	β is clarification question for α
COMMENT	β provides an opinion or evaluation for the content of α
NARRATION	The main eventualities of α and β occur in sequence
CONTINUATION	β and α elaborate on the same topic
CONTRAST	α and β have similar semantic structures, with contrasting content
PARALLEL	Same as CONTRAST, with echoing content, maybe ellipsed
RESULT	The main eventuality of α is the direct cause of β
BACKGROUND	β provides some stage setting for what happens in α
CONDITIONAL	Typically: if α, then β
ALTERNATION	Typically: α, or β

Table 2.3: SDRT relations. α and β designate respectively the first and second arguments, in textual order, of the relation.

2.2 Discourse parsing

The previous section showed how we can describe, as accurately as possible, the discourse structure of texts. However, the next step is to build them automatically. Works dedicated to this task, *discourse parsing*, have been much more recent than the work on representation. We will review in this section the progression and challenges of the domain, starting from low-context environments to our current rich domain, dialogue.

[2]Also named FOLLOW-UP QUESTION

2.2.1 Cue-based parsing

Here is again the John & Max example:

(2.13) [Max fell]$_{a1}$, [because John pushed him]$_{a2}$.

(2.14) [Max fell]$_{b1}$. [John pushed him]$_{b2}$.

In both examples, EDUs 1 and 2 are linked by a causal relationship (in RST and SDRT, EXPLANATION). In the first example, this discourse relation is hinted by the word *because*; as in PDTB we call this kind of hint a *discourse marker*.

Marcu (1997), pioneering the field of discourse parsing, attempts to recreate full RST structures from discourse markers. Marcu argues that markers are consistently used by humans throughout text; that they occur frequently enough to infer the structure of a text from them alone; that the semantics of the markers are consistent with the semantics of the components they link.

Marcu points out the ambiguity of markers, with respect to the relations they convey, and their reach in the text. This example is given:

(2.15) [*Although* discourse markers are ambiguous]$_1$, [one can use them to build discourse trees for unrestricted texts:]$_2$ [this will lead to many new applications in natural language processing.]$_3$

Does the ELABORATION relation cued by the colon links EDUs 2 and 3, or 1 and 3? In the proposed parser, the second option is ruled out due to the CONCESSION relation cued by *Although*: EDU 2 is the Nucleus of the $1-2$ span, and by the Nuclearity Principle, if a relation should link EDUs $1-2$ and 3, it must hold between the Nuclei of the span, i.e. 2 and 3.

Marcu (2000) details the parsing method. The parser uses a wealth of information regarding the behavior of discourse markers, among which:

- if they appear before, between, or after the two spans they link;
- the boundaries of the spans;
- the textual types of the spans (some markers link clauses together, other whole paragraphs);
- the rhetorical status (Nucleus or Satellite) of the spans;
- the rhetorical relations associated with the marker.

Many other fields are included in the analysis of markers, created from careful review of annotated examples. This enables Marcu to create a shallow analysis of texts, procedurally generate their segmentation, and hypothesize relations between spans. The possible structures verifying the set of hypotheses are then enumerated exhaustively, and assigned a weight (privileging balanced trees). The structure of highest weight is then returned as the parsed rhetorical structure of the text.

2.2.2 Implicit relations

As stated earlier, the entire process in Marcu (2000) relies on the presence of explicit discourse markers to identify relations and their spans. However, this hypothesis doesn't hold in the general case.

Marcu and Echihabi (2002) refer to the then-recent RST Discourse TreeBank (Carlson et al., 2003)[3], observing that less that a third of the CONTRAST and EXPLANATION-EVIDENCE relations in the corpus were marked by a cue phrase. Those two relations being extremely distinct semantically, the ambiguity caused by the absence of markers has to be resolved by other methods. The NLP field doesn't have access to robust semantic interpreters and knowledge bases powerful enough to infer from example 2.4 that the fall of Max was caused by John pushing him.

Marcu and Echihabi study how to disambiguate between rhetorical relations. Their novel approach is to consider discourse markers as *additional* material for the semantic parsing of a pair of span; in other words, that the spans retain semantic cues of rhetorical relations even if the explicit markers are removed. Consider the following sequence:

(2.16) John is *good* in math and science.

(2.17) Paul *fails* almost every class he takes.

The two sentences are linked by a CONTRAST relation. The two words *good* and *fails*, as a pair, are good indicators of contrasting statements. The authors hypothesize, I quote: "that lexical item pairs can provide clues about the discourse relations that hold between the text spans in which the lexical items occur." In order to predict, from statistical methods, which lexical pairs imply which relation, one would need a large corpus of annotated pairs, not available at the time.

Marcu and Echihabi choose a restricted set of rhetorical relations to disambiguate between; namely, CONTRAST, CAUSE-EXPLANATION-EVIDENCE, CONDITION, ELABORATION, and the default relation NONE OF THE ABOVE. The

[3]The initial release of the paper and the associated corpus dates from 2001.

relations are clearly distinct semantically, defining coarse groupings of usual label (the coarse label CONTRAST encompasses ANTITHESIS and CONCESSION, for example). This choice enables the authors to build a low-noise dataset from cue phrases present in unannotated large corpora. Using a collection of nearly 43 million sentences gathered from various sources and various extraction patterns (see Table 2.4), a corpus of millions of automatically annotated pairs of spans is created.

Label	Instances	Example pattern
CONTRAST	3,881,588	[BOS ...][but ... EOS]
CAUSE-EXPL.-EV.	889,946	[BOS ...][because ... EOS]
CONDITION	1,203,813	[BOS If ...][then ... EOS]
ELABORATION	1,836,227	[BOS ... EOS][BOS ... for example ... EOS]

Table 2.4: Patterns for automatic extraction of related pairs. BOS and EOS stand for *Beginning* and *End Of Sentence*, respectively.

The markers present in the annotated pairs are then removed, and the data is provided to a Naive Bayes method, which gives the most probable relation from the words in a pair of text spans:

$$\begin{aligned} r^* &= \operatorname*{argmax}_{r_k} P(r_k|W_1, W_2) \\ &= \operatorname*{argmax}_{r_k} (\log P(W_1, W_2|r_k) + \log P(r_k)) \\ P(W_1, W_2|r_k) &= \prod_{(w_i,w_j)\in W_1\times W_2} P(w_1, w_2|r_k) \end{aligned}$$

where (r_k) are the relation labels, W_1 and W_2 are the two word sequences of the spans. Probabilities $P(w_1, w_2|r_k)$, linking word co-occurences to labels, are computed over the corpus by a maximum likelihood estimator.

For the task of classifying pairs of spans into the six categories cited above, this model obtains an accuracy of 49.7%. Two-way classifiers were also tested, with greater performance. For instance, two-way disambiguation between CAUSE-EXPLANATION-EVIDENCE and ELABORATION attained 93% accuracy (their best result).

Sporleder and Lascarides (2005) use a similar approach, disambiguating from SDRT relations, namely CONTRAST, RESULT, EXPLANATION, SUMMARY and CONTINUATION. Their training corpus was, as well, built from a compilation

of written text corpora, mainly from the new domain. The resulting set of extracted pairs was smaller, from less than 2,000 examples for the CONTINUATION relation, to around 50,000 for CONTRAST.

Instead of relying on word co-occurrences for probability estimation, the authors relied on a set of shallow features extracted from span pairs:

- the length of the spans;
- lexical features, such as the string of lemmas contained in the spans, the overlap of lexicon between the two spans, and the WordNet (Miller, 1995; Fellbaum, 1998) class of lemmas;
- part-of-speech features, such as the string of POS tags of the spans;
- temporal features, classifying verbal complexes along five criteria (Lapata and Lascarides, 2004);
- syntactic features, extracted from parse trees;
- cohesion features, from the distribution of pronouns and the use of ellipses.

Their 5-way classifier, using a model similar to Marcu and Echihabi (2002), attained 33.96% accuracy, with a smaller dataset.

For another example of pattern-based extraction of training data, Saito et al. (2006) use Japanese phrasal patterns as indicators of rhetorical relations.

2.2.3 PDTB connectives

Many works focused on predicting local discourse relations use the Penn Discourse TreeBank (Miltsakaki et al., 2004; Prasad et al., 2008), which incidentally is the largest corpus dedicated to local structures of discourse.[4]

Wellner and Pustejovsky (2007) propose a method to identify the arguments of PDTB discourse connectives. In the PDTB, explicit discourse relations are annotated with the connectives themselves, so that parsing explicit relations amount to find the spans linked by the connectives. They use a variety of features, drawing from diverse parsers:

- *Baseline features* describing the location of the connective, the (candidate) arguments themselves;

[4]If this sounds tautological, we want to stress that the publication of new reliable annotated datasets, (not only in discourse parsing but in natural language processing as a field), is a critical driving force behind many new advances. Annotations are expensive.

- *Constituency features* based on a constituent (syntactic) parse of the arguments;
- *Dependency features* based on a dependency parse of the arguments;
- *Connective features* based on the definition of the connective itself;
- *Lexico-syntactic features* detecting attribution patterns (as in *X said Y*)

With the use of probabilistic ranking models[5], the authors achieve .887 F-measure on identifying segment boundaries, and .763 F-measure when also labeling the arguments as nucleus and satellite.

Pitler et al. (2009) introduce a set of original shallow features for sense prediction. They include:

- *Polarity tags*, words in the spans indicating sentiment (such as *good, nice, awful*, etc.;
- *Inquirer tags*, same as above with various semantic classes from the General Inquirer lexicon (Stone et al., 1966);
- *Money-Percent-Num*, detecting numerical figures in text;
- *WSJ-LM*, classifying the likelihood of the span's words with respect to the relation labels;
- various other features describing the verb occurring in the spans, the first and last words of the spans, modality markers (e.g. *can, should*), preceding explicit connectives, and word pairs.

Lin et al. (2009) focus on implicit relations, which are evidently harder to extract than their explicit counterparts. Using a Maximum Entropy classifier (Berger et al., 1996), they also use a feature set with similar categories as Wellner and Pustejovsky (2007), drawing from dependency and constituency parsing, as well as word pairs, like Marcu and Echihabi (2002). In particular, they discuss the difficulties of the extraction of implicit connectives:

- *Ambiguity*: several relations, such as CONTRAST and CONJUNCTION, are very similar in syntax, lexicon and semantics. A formulation like *X, while Y* can be interpreted both ways, even if the connective *while* is explicit. An analysis of the context may disambiguate between several senses, which would require further annotation effort;

[5]We refer the reader to the paper for additional equations.

- *Inference*: as in the example *Max fell; John pushed him*, external knowledge and semantics is sometimes needed to infer discourse relation;
- *Context*: while the annotated arguments in the PDTB are enough for the interpretation of the relation, more context is sometimes needed to understand the argument themselves (and predict the relation from unannotated text);
- *World knowledge*: discussed with the example below:

(2.18) **Arg1:** Senator Pete Domenici calls this effort "the first gift of democracy".
Arg2: [but] The Poles might do better to view it as a Trojan Horse.
*(*CONTRAST *- PDTB - wsj_2237)*

Here, one has to recognize that a *Trojan Horse* is a kind of *gift*, and infer the CONTRAST relation from the negative connotation. Alongside the other difficulty classes, which are mirrored in many sub-domains of natural language processing, this illustrates the need for deeper semantic representations and access to more world knowledge.

Zhou et al. (2010) attempt to predict implicit discourse *connectives* as an intermediate step to predict discourse relations. The first task of predicting connectives uses a small set of features, and outputs a set of 60 most probable connective for a given pair of spans.[6] They propose two approaches to the second task of predicting relations: one using the predicted connectives as additional features, the other using the predicted connectives alone as features. The results vary greatly depending of the label, but consistently beat the baseline classifier following the usual approach of predicting directly the relation label from the initial features.

Rutherford and Xue (2014) use Brown clusters (Brown et al., 1992) (grouping words appearing in the same contexts) to tackle the problem of the sparsity of word pairs. Replacing each word by their cluster generates a much smaller feature set, as 3200 clusters were generated, for a vocabulary several orders of magnitude bigger. Aside from word pairs, the clusters are also used to define several new features, detecting the number of same-cluster words (or specifically nouns or verbs) present in both spans. Their experiments, using Naive Bayes classification, improve previous performance on one-against-all labeling tasks (where the goal is to determine if a pair is linked by a particular relation or not).

[6] At this point, we'd like to point out that borrowing features from preceding literature is a common occurrence, which explains the succinct mention of the feature sets, which are no longer the main focus of the publications.

Braud and Denis (2015) expand on their work, using alternative word and segment representations as features. They describe three ways to represent a word with vectors;

- one-hot, equivalent to word count methods, where each word has its own dimension and is counted separately;
- cluster-based one-hot, as in Rutherford and Xue (2014), where one dimension corresponds to one cluster;
- dense real-valued, low-dimensional vectors, where dimensions correspond to latent features of words, often learned through neural models (Bengio et al., 2003) or distributional analysis;

The authors' next step is to represent whole spans with vectors before using them as training instances. They describe multiple methods to get there from the vector representation of the component words:

- by considering the word vector of the span's syntactic head only, or all words (and eventually normalizing the results to compensate for the word count of the span);
- by concatenating the vectors from both spans, or taking their outer product (in the one-hot case, this amounts to have one dimension per word pair);

These methods, along with the choice of one-hot or dense representation of words, combine into numerous possible representations for pairs of spans. Using a Maximum Entropy classifier trained on vector features, heads of spans, and commonly used other features, the model reaches a similar performance as Rutherford and Xue (2014).

2.2.4 Full structure parsing

While the above cited works explore in depth the prediction of local discourse relations, comparatively few works attempt to predict the full discourse structure of a text. Early works, without any global optimization, include Marcu (2000), relying on explicit cues, which are in minority in the RST-DT corpus.

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 labeling of the links between discourse units with discourse relations. Many recent works take segmentation for granted, as reliable methods exist for the task.

Soricut and Marcu (2003) explore discourse parsing for the restricted scope of a sentence, using RST-DT as a corpus. They detail the two challenges of the task. First, discourse segmentation; much like Wellner and Pustejovsky (2007) sought to find the spans of text involved in discourse relation, to parse a text one needs to split it in *Elementary Discourse Units*, forming the basic component of the structure. The segmenter proposed by the authors uses a probabilistic likelihood model to predict unit boundaries, using lexical and syntactic features.[7]

Once the text has been segmented, the authors use another probabilistic model to build the structure. The objective is binary trees instead of general RST trees, as 99% of the nodes in the RST-DT corpus are binary (the resulting model is made simpler by this choice). In the paper's formalism (which will be reused in Joty et al. (2012)), an RST relation is written as a tuple $R[i, m, j]$, where R is an RST label augmented by nuclearity (that is, which of the relation's arguments are Nucleus or Satellite), holding between the two spans containing EDUs i through m, and $m + 1$ through j. The parser the uses a bottom-up dynamic programming algorithm to determine the subtrees of highest probability, merging adjacent spans together repeatedly until all spans are merged, forming a binary tree.

One specific feature used in their model concerns the notion of *dominance set*, which describe where and how two adjacent EDUs are linked in the syntactic tree. For all EDUs except the one containing the root of the tree, the syntactic head of the EDUs will have a parent belonging to another EDUs. The direction of this link is the *dominance relationship* between the two EDUs. Dominance sets are used to filter out irrelevant elements while computing the probability of a subtree.

Le Thanh et al. (2004) use a cue-based segmenter and bottom-up parser to build RST trees. Interestingly, they compute the accuracy of their parser on seven distinct levels: discourse unit boundaries, local attachment only, nuclearity role of spans, full discourse relation (attachment and label); the three latter being evaluated once at the sentence level, once at the text level.

Baldridge and Lascarides (2005) choose to study dialogue parsing; more specifically, appointment scheduling dialogues from the Redwoods corpus (Oepen et al., 2004). They re-annotated the corpus using a restricted version of SDRT, encoded into trees so their model could use statistical techniques from sentential parsing, namely Probabilistic Context Free Grammars.

Sagae (2009) pioneered the technique of *transition-based discourse parsing*, building an RST tree by the shift-reduce method. Starting with a stack of subtrees

[7]Work on segmentation not being the focus of this review, we refer the reader to the paper for more details.

containing the first EDU of a text (as an atomic subtree), the parser performs one of three kinds of action: *shift*, where the next EDU is pushed on the stack; *reduce-left-LABEL* and *reduce-right-LABEL*, where the two topmost subtrees are joined together with the relation *LABEL* to form a bigger subtree (so that each label corresponds to two actions). Whether the head of the newly created subtree is the left or right one depends on the aptly named *left* or *right* version of the action.[8] The action to be performed by the parser is determined through an averaged perceptron,[9] using basic lexical and syntactic features. The parser achieves a performance of .445 F-measure for full discourse tree creation, evaluated on RST-DT.

Subba and Di Eugenio (2009) also use transition-based parsing, with an Inductive Logic Programming (ILP)[10] to create a ruleset determining which action the parser should perform. The rules are built on lexical features as well as similarity features based on the author's previous work (Subba et al., 2006). Here is an example of one of their generated rules:

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

duVerle and Prendinger (2009) introduce a new method, Support Vector Machines (SVM) (Vapnik, 1995), to estimate the probability of subtrees (using the same greedy bottom-up tree-building algorithm as Soricut and Marcu (2003)). SVM is a machine learning technique well-suited to classification problems involving high-dimensional feature spaces, which had yet to be applied to discourse parsing, as SVM exclusively performs binary classification, and adaptations were necessary. The authors thus use two classifiers, one for detecting whether two adjacent subtrees are directly connected or not, and another for predicting the relation labels (using a multi-class variant of the model, per Crammer and Singer (2001)).

Hernault et al. (2010) expand on their own work,[11] adding a discourse segmenter, which also uses SVM to detect the presence of EDU boundaries, with features inspired from Soricut and Marcu (2003). The proposed full parser achieves a

[8] Another action, *reduce-unary-LABEL*, which takes only one argument subtree, is also described but unused when creating binary RST trees.

[9] A linear binary classification method, of the neural network class.

[10] Not to be confused with *Integer Linear Programming* (also ILP), an entirely different technique used in our own work and described later on.

[11] Which is duVerle and Prendinger (2009), as they're also co-authors of this paper.

performance of .473 F-measure for full discourse tree creation, evaluated on RST-DT.

Feng and Hirst (2012) expand on Hernault et al. (2010), predicting RST structures, incorporating features from the work of Lin et al. (2009) in PDTB relation parsing, as well as their method of feature selection.

Joty et al. (2012) focus on sentence parsing. They introduce, in turn, a new method to estimate the probability of subtrees: Dynamic Conditional Random Fields. DCRFs (Sutton et al., 2007) is a *structured* machine learning technique suited to predict sequences of items. Where regular models accept features describing two spans, merged into a single training instances, DCRFs allow the authors to input an *arbitrarily long sequence* of spans to the model, each having their own features, so that the output for a given span is dependent on the rest of the sequence.

The model enumerates all the possible span combination for a given sequence. For three EDUs, the possible groupings are `([1], [2], [3])`, `([1], [2-3])`, `([1-2], [3])`, `([1-2-3])`. For every span combination, a DCRF is generated, of the following form[12]:

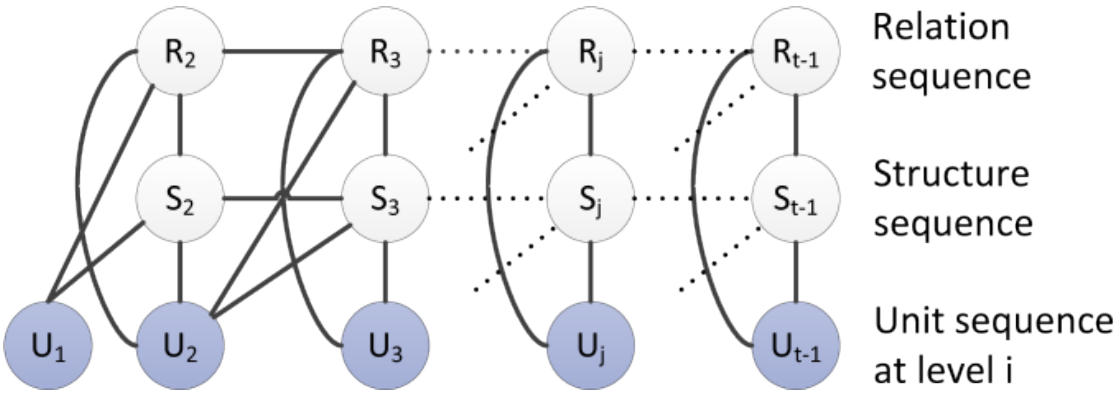


Figure 2.6: Structure of a CRF model for a sequence of spans

The unit sequence at the bottom is the sequence of spans; the S_i nodes output probabilities whether spans U_{i-1} and U_i are linked by a relation; the R_i nodes (of which there are actually several layers, one per label) output probabilities for the relation labels. In conjunction, over all possible combinations of spans, the DCRFs produce the probabilities of the constituent $R[i, m, j]$.[13] These probabilities

[12]Picture borrowed from Feng and Hirst (2014)

[13]A relation of label R, between the spans i through m and $m + 1$ through j. See the above paragraph on Soricut and Marcu (2003).

are then used in a CKY-like[14] bottom-up algorithm, which, unlike the greedy algorithm of Hernault et al. (2010), returns the globally optimal tree given the input constituent probabilities.

Joty et al. (2013) expands the method on multi-sentential parsing. The higher number of EDUs makes the computation of all combinations of spans impractical. The paper introduces a new CRF structure designed to predict the attachment of pairs of adjacent spans only (which are far fewer). In order to parse a whole text, the parse trees of pairs of adjacent *sentences* are built, so that every non-terminal sentence belongs to two parse trees. The trees are then combined to produce the final parse tree of the text.

Joty and Moschitti (2014) expand *again* their method by generating the k-best parses for multi-sentential texts, then using *tree kernels* to re-rank the parses. The kernels enable trees to be used directly as input for the following SVM classification task: *should the pair of parses* (T_i, T_j) *be re-ranked?*. The results are then combined to find the new best parse.

Joty et al. (2015) finally adds a segmenter (which uses a Maximum Entropy model) to their parsing framework to complete it.[15]

Muller et al. (2012b), precursor to our own research, parse the French-language ANNODIS corpus, comprising newspaper and Wikipedia articles annotated in SDRT fashion. Their framework uses a probabilistic (Maximum Entropy) model of local relations based on shallow features; they experiment with two heuristic-based approaches, the Maximum Spanning Tree method and A* search, to obtain a *globally optimized* structure. Both outperform the replicated greedy approach of Hernault et al. (2010).

Li et al. (2014) use the RST-DT corpus, Margin-Infused Relaxed Algorithm (McDonald et al., 2005) for learning feature weights, and create tree structures using the Eisner algorithm (Eisner, 1996) as well as the MST algorithm as decoders.

While it doesn't predict discourse relations *per se*, we also mention Elsner and Charniak (2010), exploring the task of *disentangling* IRC[16] chats, which often involves groups of people carrying multiple discussions at the same time through the same channel. Their work isn't the first in the domain, and we invite the

[14] Referring to the Cocke–Younger–Kasami algorithm, described in Jurafsky and Martin (2014).

[15] They also give it a little name: CODRA. The framework from Hernault et al. (2010) was named HILDA, and before that Soricut and Marcu (2003) had SPADE. It may be a tradition.

[16] Acronym for *Internet Relay Chat*, a protocol for text-based online discussion.

reader to follow their citations. Their model uses a maximum entropy classifier with shallow features to detect whether pair of utterances are part of the same discussion thread, then aggregates the results using clustering algorithms.

2.2.5 Observations on the state of the art

We recapitulate a few overarching elements from the progression of the field of discourse parsing.

Shallow features are useful in parsing. The feature sets vary, of course, between the publications, but simple features such as syntactic heads and labels, presence of words of various lexicons, part-of-speech tags, are frequently used successfully throughout the literature. Analysis of features is no longer frequent, and seemingly reserved to long-form papers. For instance, Hernault et al. (2010) provide a list of the most influential features by weight, in their SVM linear kernel for predicting whether two spans are linked or not. Table 2.5 displays the first ten items of their list.

Feature	**Weight**
Both spans belong to the same sentence	4.118836
Size of span over sentence in EDUs	3.582545
Distance of the left span to beginning of sentence in EDUs	-3.437157
Common ancestor's POS tag is 'PRN'	-2.911269
Dominating node's lexical head is 'which'	-2.668148
POS tag of the right span's last token is '.'	2.636921
Size of left span over sentence in tokens	-2.341654
Size of both spans over sentence in tokens	-2.222655
Left and right span belong to the same sentence	-2.217709
POS tag of the left span's last token is '.'	2.170483

Table 2.5: Most weighted features of the linear kernel for attachment prediction of Hernault et al. (2010).

Local parsing of relations still isn't robust. The use of increasingly sophisticated techniques to reduce the feature space of learning models, such as Brown clusters Rutherford and Xue (2014) or word embeddings Braud and Denis (2015), accelerates the training of the systems, allowing them to work on bigger datasets (which are already scarce). However, they don't address directly the problems

described in Lin et al. (2009), calling for better semantic models of context and knowledge representation.

Most works focus on monologue. Before the publication of the *Settlers* corpus, discussed in chapter 4, RST-DT and PDTB were the two most prominent corpora for the study of discourse, both containing only monologues and driving the majority of the above-cited research, one notable exception being Baldridge and Lascarides (2005) (focused on dialogue). We'll develop on the key differences between monologue parsing and dialogue parsing in section 4.2.

2.3 The Right Frontier Constraint

2.3.1 History

Many theories of discourse structure posit a Right Frontier Constraint (RFC) on discourse attachment (Polanyi and Scha, 1984; Polanyi, 1985; Webber, 1988). The RFC restricts the attachment of newly processed units of a discourse to a small subset of the units in the structure already constructed for some portion of the discourse. The motivating hypothesis behind the RFC is that discourse structure plays a major role in controlling salience. A coherence relation R inferred between two bits of a discourse d will have a particular effect on the shape of the overall tree or graph used to represent d's structure in a way determined by the semantics of R and the discourse theory in use. Relations thus determine what nodes are found along the tree or graph's Right Frontier (RF), a set that evolves dynamically as a discourse proceeds. The RF *constraint* captures the observation that new utterances are normally attached to these nodes, which are predicted to be the most salient.

The RFC constrains semantic phenomena like anaphora and topic, as antecedents for most anaphoric expressions and ellipses are hypothesized to be found along the RF (Polanyi, 1985; Webber, 1988; Asher, 1993). It is also potentially helpful for discourse parsing: restricting attachments to units on the RFC considerably reduces the search space for attachments for discourse units and thus has the potential to improve inter-sentential attachment scores, which are in general much lower than scores for intra-sentential attachment (Joty et al., 2015). Note, however, that the RFC rarely on its own determines attachment, and it can be violated in certain discourse configurations (Asher, 1993; Prévot and Vieu, 2008), though violations are rare in our corpus study (cf. section 5.1.4). The RFC is a defeasibly necessary but not sufficient constraint.

More importantly, the RFC is practically the only structural constraint on discourse attachment that takes the overall *structure* into account. Most discourse

parsing models optimize probabilities for attachments over pairs of elementary discourse units, based on features like textual distance or grammatical or lexical properties of the paired elements. While local features are useful, discourse parsing performance lags behind syntactic parsing, because it does not use global features, in the way syntactic methods have done since Collins and Duffy (2002). The RFC is just such a global feature: it says the overall structure of the discourse graph has to have a certain shape. Because of data sparseness and our current limitations to supervised learning, it is infeasible to learn probabilistic global constraints like the RFC from the data directly. So defining an appropriate RFC via symbolic methods is a necessary step to improve discourse parsing.

The RFC has in practice been developed for, and tested on, monologue, generally in the form of newspaper texts (Afantenos and Asher, 2010). It is expected to be helpful as a constraint on multilogue as well, though important differences between multilogue and monologue prevent a trivial extension of standard RFC definitions. In monologue, a speaker is uniquely responsible for the information presented in the discourse, and the RFC is a constraint on the way that information should be presented. In dialogue, we deal not only with how speakers present information but also how they pick up on information presented by others. One speaker might make multiple points, but her respondent might pick up on just one, or ignore them all. Or one or more respondents might wish to discuss multiple points simultaneously, introducing multiple conversation threads.

The RFC is related to projectivity in parsing (Nivre, 2003). Like projectivity, RFC compliance is a property of a graph with respect to textual order, and like projectivity, the RFC rules out crossing dependencies (relative to textual order) except in special cases. Unlike projectivity, however, the RFC depends on a semantic distinction between subordinating and coordinating relations, and a distinction between CDUs and EDUs. Projectivity and the RFC are thus not equivalent even on trees.

The RFC has been a topic of interest in theoretical work on discourse structure for a long time. But to our knowledge, we are the first to study how it fares for multilogue on a large discourse annotated corpus. With regard to empirical work on discourse parsing, Afantenos and Asher (2010) demonstrate the potential of this constraint, but we are not aware of any actual parsing results with the RFC for monologue or dialogue. They also conducted an empirical study on RFC for monologue.

2.3.2 Formal definition

In general, when an utterance u is made, the content of the utterance immediately prior to u will be highly salient, but other contents might be salient as well. A speaker might linger on a topic—elaborating on it, providing background on it,

or explaining it and so on. In such a case, the point that is being elaborated on or explained, etc. will remain salient, and potentially form a chain of salient and accessible contents underneath it.

On the other hand, when a speaker, say, lists a series of attributes or describes a sequence of events, the most recently described attribute/event will be more salient than the previously described ones, rendering the latter inaccessible to later utterances. Thus in (2.19), the content of π_1 is inaccessible to that of π_3—we cannot infer the sequence $\pi_1 + \pi_3 + \pi_2$, even though that would yield a more coherent discourse (without further context).[17]

(2.19) Rose dumped the cookies on the floor.$_{\pi_1}$ (So) She was sent to her room.$_{\pi_2}$ (And) She drew all over the kitchen wall.$_{\pi_3}$

If we reverse the order of π_2 and π_3, as in (2.20), we can group Rose's two acts together, as desired. What's more, while π'_1 alone is inaccessible to π'_3, the fact that π'_2 clearly describes an event in a series of related events makes the group $\pi'_1 + \pi'_2$ salient and accessible. That is, we understand Rose's being sent to her room as the result of both acts, not just of the more recently described one.

(2.20) Rose dumped the cookies on the floor.$_{\pi'_1}$ (And) She drew all over the kitchen wall.$_{\pi'_2}$ (So) She was sent to her room.$_{\pi'_3}$

To make this precise, let's consider the RFC as defined in Segmented Discourse Representation Theory. In SDRT, the structure for a discourse d is modelled as a rooted spanning directed acyclic graph, called an SDRS, $G = (V, E_1, E_2, Last)$. V is the set of *elementary discourse units* (EDUs; labeled $\pi_0, ..., \pi_n$) and *Complex Discourse Units* (CDUs) in d, where an EDU is a clausal or sub-clausal unit and a CDU is a collection of EDUs (and possibly other CDUs) that together serve as an argument to a discourse relation. $E_1 \subseteq V \times V$ is the set of edges or labeled discourse attachments between elements of V. $E_2 \subseteq V \times V$ is the parenthood relation that relates CDUs to their component DUs. We write $e(\pi_x, \pi_y)$ when e is an edge with initial point π_x and endpoint π_y. *Last* is the last EDU in V, following the linear ordering of EDUs determined by their order in d. An SDRS is "spanning" in that all elements of V other than the root have at least (and possibly more

[17]Eliciting intuitions about examples like (2.19) is a delicate matter. While rhetorical theories hold that discourse structure and coherence are intimately related, this does not mean that other factors, such as intonation and word choice, do not affect coherence. In (2.19), it is important to read the example with a normal intonation. Were a speaker to preface π_3 with *and* and pronounce *and* with a certain intonation, it would be clear that she wanted to retroactively add π_3 to the list of reasons why Rose was sent to her room, i.e. π_3 could attach to π_1. However, the special intonation would arguably be a signal that the speaker wanted to return to a less salient point.

than) one incoming edge:

$$\forall \pi_x \in V : (\pi_x \neq \text{ROOT} \rightarrow \exists \pi_v \in V : ((\pi_v, \pi_x) \in E_1))$$

The set E_1 can contain two types of edges, *coordinating* and *subordinating*. Relations such as EXPLANATION, ELABORATION, and BACKGROUND—in which the second argument extends the discussion about the first—are represented with subordinating (vertical) edges. Relations such as CONTINUATION, NARRATION, and RESULT—in which the second argument shuts off the accessibility of the first—are represented with coordinating (horizontal) edges. Suppose we prefix (2.20) with π_0, *We've been having a rough time*, so that π'_1–π'_3 elaborates on π_0. π_0+(2.20) would yield the graph $G_{\pi_0+(2.20)}$:

- $V = \{\pi_0, \pi'_1, \pi'_2, \pi'_3\}$
- $E_1 = \{\langle \pi_0, C_1 \rangle, \langle \pi'_1, \pi'_2 \rangle, \langle C_0, \pi'_3 \rangle\}$
- $E_2 = \{\langle \pi'_1, C_0 \rangle, \langle \pi'_2, C_0 \rangle, \langle C_0, C_1 \rangle, \langle \pi'_3, C_0 \rangle\}$
- *Last*$= \pi'_3$.

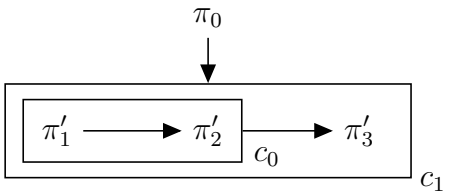


Figure 2.7: Graph of π_0 + (2.20)

For monologue, a node π_x is on the RF of a graph G, i.e. $\text{RF}_G(\pi_x)$, if either π_x is *Last*, or π_x is related to *Last* via a series of subordinating (*Sub*) edges, or π_x is a CDU that includes a node in RF_G. Formally, let $G = (V, E_1, E_2, Last)$ be a discourse graph.

$$\begin{aligned} \forall \pi_x, \pi_y, \pi_z \in V \quad \text{RF}_G(\pi_x) \iff & \ \pi_x = Last \\ & \vee (\text{RF}_G(\pi_y) \wedge \exists e \in E_1, e(\pi_x, \pi_y) \wedge Sub(e)) \\ & \vee (\text{RF}_G(\pi_y) \wedge \exists e \in E_2, e(\pi_x, \pi_y)) \end{aligned}$$

So the RF of $G_{\pi_0+(2.20)}$ is $\{\pi'_3, C_1, \pi_0\}$. Note that the RF is updated dynamically each time a new EDU is processed; the RF for (attachment of) an EDU π_n will be determined by the graph $G_{\pi_0-\pi_{n-1}}$. The RF for a CDU $\pi_m \ldots \pi_n$, $m < n$, is the RF for π_m.

Chapter 3

Background: argumentation

While the study of discourse has progressed closely to the study of semantics and syntax, the study of argumentation began with the very fundamentals of logic, from Aristotle:

- All humans are mortal.
- Socrates is human.
- Therefore, Socrates is mortal.

Logical arguments and their validity have been studied for millenia, forming the basis of argumentation theory. In this chapter, we will give the reader an overview of the main concepts relevant to our study of argumentation, as well as a quick review of the effort directed towards the automatic extraction of argumentative elements.

3.1 The building blocks of argumentation

This description follows the formalism of Peldszus and Stede (2013), which is used in our work on argumentation. While there exists an important number of detailed descriptions of the following concepts in the literature, their core meaning stems from the same basic definitions.

Claims At the center of argumentation structure is the *claim*, the assertion that something is true, or false. A claim doesn't have an embedded truth value. If one says *John is mean*, this claim could be true or false according to the belief of the listener; a claim is only a *declaration* of truth (or falsehood), which may itself be interpreted. A claim can eventually be formalized as a logical predicate.

Support The usual goal of a claim is to increase the belief of the receiver in that claim.[1] A *supporting claim* has the goal of increasing belief in the supported claim, by virtue of being believable itself, and being relevant to the original claim. In the previous example, the claims *All humans are mortal* and *Socrates is human* both support the claim *Socrates is mortal.*

Support takes several forms. A claim can, alone, support another claim; a *set* of claims can support another claim, with all the parts being necessary. In the previous example, *All humans are mortal* isn't enough of a claim to support alone *Socrates is mortal.* A claim can support another claim that supports another claim, and so on.

Attacks An *attacking claim* has, symmetrically, the goal of *decreasing* belief in the attacked claim, by the very same virtues of being believable itself, and being relevant to the original claim. While attacks can simply target the original claim or its supporters (i.e. a *rebuttal*), an *undercut* attack can challenge the *relevance* of a particular support:

- A: Carthage is threatening Rome.
- A: Therefore, Carthage must be destroyed.
- B: Maybe we could negotiate with the threat instead.

Here, B doesn't deny that Carthage is a threat, but attacks the *inference* that it warrants a swift obliteration.

Attacks (and supports) can also target implicit claims:

- A: If the bill passes, riots will occur.
- B: Don't worry, the Senate won't let it pass.

Here, with the only explicit claim by A is *if bill passes then riots*, B attacks indirectly the implicit claim *riots will occur* by attacking another implicit claim, *the bill will pass.*

These building blocks of claims, support, attack, counter-attack, undercut, implicit content, can be combined in every way possible, which have been represented in various theories focusing on particular aspects of argumentation, with variable formalism.

[1] Leaving out sarcasm, which highlights the difference between the semantic meaning of discourse and its intended meaning, and introduces a great deal of complexity in argumentation theories. We'll ignore it for now.

For instance, Apothéloz et al. (1993) describe four different ways to attack a claim with supporting arguments through counter-argumentation, which specifically involves the supporting arguments, instead of moving on another relevant topic. In all cases, a different part of the argument is attacked. Consider the following examples:

(3.1) A: This movie was awful. The CGI was too visible.

a. B: I didn't even notice the CGI.

b. B: But it had a great plot!

c. B: Plenty of good movies have visible CGI.

d. B: That's what makes it good.

The four answers counter respectively: the *plausibility* of the reason (attacking the supporting claim); the *completeness* of the reason (attacking the conclusion directly with another claim); the *relevance* of the reason (undercutting the support by attacking the implicit inference); the *argumentative orientation* of the reason (undercutting the support by reusing the same argument, this time as an attack).

3.2 Argumentation structures

This section is a review of some major works centered on the task of giving argumentation a structure, beyond the classical formalism of premises and conclusions.

Toulmin (1958), in his very influential early work, describes the practical roles of arguments in persuasive texts. Initially based on an analysis of courtroom arguments, Toulmin's layout identifies six different components of an argumentative text, the first three of which are always encountered:

- **Conclusion**: the core claim of the argument, supported by the rest of the text;
- **Ground**: a base fact (which doesn't need backing), evidence that supports a claim ;
- **Warrant**: a statement describing how the Conclusion can be inferred from the Grounds;
- **Backing**: additional support for Warrants, in case they're not convincing enough;
- **Rebuttal**: a statement describing how the argument may be undermined;

- **Qualifier**: giving additional information about the force of the claim.

This separation has been discussed extensively in following works. Some of the main criticisms expansions of the model are:

- a Warrant can be viewed as the Conclusion of its own argument, creating a nested argument structure;
- the separation between Ground and Warrant isn't always clear. Both are claims supporting the conclusion, but ground evidence itself, being an interpretation of reality, can also require backing;
- only the Rebuttal component may contain claims undermining the Conclusion. In general the role of an eventual opponent is not properly represented, which limits the usefulness of the framework in debates.

Van Eemeren and Grootendorst (1992) initiate the *pragma-dialectical* theory of argumentation, treating it as a complex speech act that obeys to informal rules to remain focused and non-fallacious. A debate is viewed as an exchange of standpoints, i.e. positive or negative stance toward a particular claim, and arguments to support one of the two. When Gricean maxims (Grice, 1978) attempt to describe the underlying assumptions of the participants in a conversation, pragma-dialectics define for debaters the following *rules for a critical discussion*, here informally abridged:

- *Freedom*: Advancing standpoints or casting doubt on them is always permitted;
- *Burden of proof*: A party must defend its own standpoints on request;
- *Standpoint*: An attack must target a standpoint actually advanced by the other party;
- *Relevance*: Defense of a standpoint must relate to the standpoint;
- *Unexpressed premise*: If a party leaves a premise implicit, they cannot deny it; nor they can force a premise onto the other party;
- *Starting point*: The premises accepted as the starting point of the debate shall not be altered;
- *Argument scheme*: A standpoint cannot be conclusively defended without appropriate argumentation;
- *Validity*: Arguments should be valid, provided potentially unexpressed premises;
- *Closure*: A failed defense should result in the retraction of a standpoint; a conclusive defense should result in the retraction of the attack;
- *Usage*: Parties should remain clear and unambiguous in their arguments, and conversely be accurate in the interpretation of the opponent.

Breaking one or several of those rules induce *fallacious reasoning*, of which the authors describe many varieties. Threatening the opponent, for example, is a basic violation of the first rule, as putting pressure on the other party hinders their freedom to cast doubt.

The structure implied by this framework is a tree of standpoints and arguments directly linked by attacks and supports.

Freeman (1991), updated by Freeman (2011), proposes a generic structure of argumentation, where claims are separated by their stance with respect to the core claim of the argument. The text can thus be viewed as an exchange between a proponent and an opponent view, attacking or undercutting the moves of the other side and supporting their own, creating the structure of a graph.

Peldszus and Stede (2013) synthesize this view in the formalism which basic concept were presented above, enabling the annotation of argumentative texts in a lightweight fashion.

In more detail, the formalism posits that the argumentative text has a central claim, which the author can back up with statements that are in a SUPPORT relation to it; this is a transitive relation, leading to "serial support" in Freeman's terms. A statement can also have multiple SUPPORTs; these can be independent (each SUPPORT works on its own) or linked (only the combination of two statements provides the support). Also, the scheme distinguishes between "standard" and "example" support, whose function originates from providing an illustration, or anecdotal evidence.

When the text mentions a potential objection, this segment is labeled as bearing the role of "opponent's voice"; this goes back to Freeman's insight that any argumentation, even if monological, is inherently dialectical. The segment will be in an ATTACK relation to another one (which represents the proponent's voice), and the scheme distinguishes between REBUTTAL (denying the validity of a claim) and UNDERCUT (denying the relevance of a premise for a claim). When the author proceeds to refute the attack, the attacking segment itself is subject to a REBUTTAL or UNDERCUT relation.

The atomic components of such an analysis are Argumentative Discourse Units, which often are larger than EDUs: multiple discourse segments can play a common argumentative role. In such cases, the EDUs are linked together by a meta-relation called JOIN.

For illustration, here is a short sample text, with its analysis shown in Figure 3.1.

Should health insurers pay for alternative treatments?

Health insurance companies should naturally cover alternative medical treatments. Not all practices and approaches that are lumped together under this term may have been proven in clinical trials, yet it's precisely their positive effect when accompanying conventional 'western' medical therapies that's been demonstrated as beneficial. Besides many general practitioners offer such counselling and treatments in parallel anyway - and who would want to question their broad expertise?

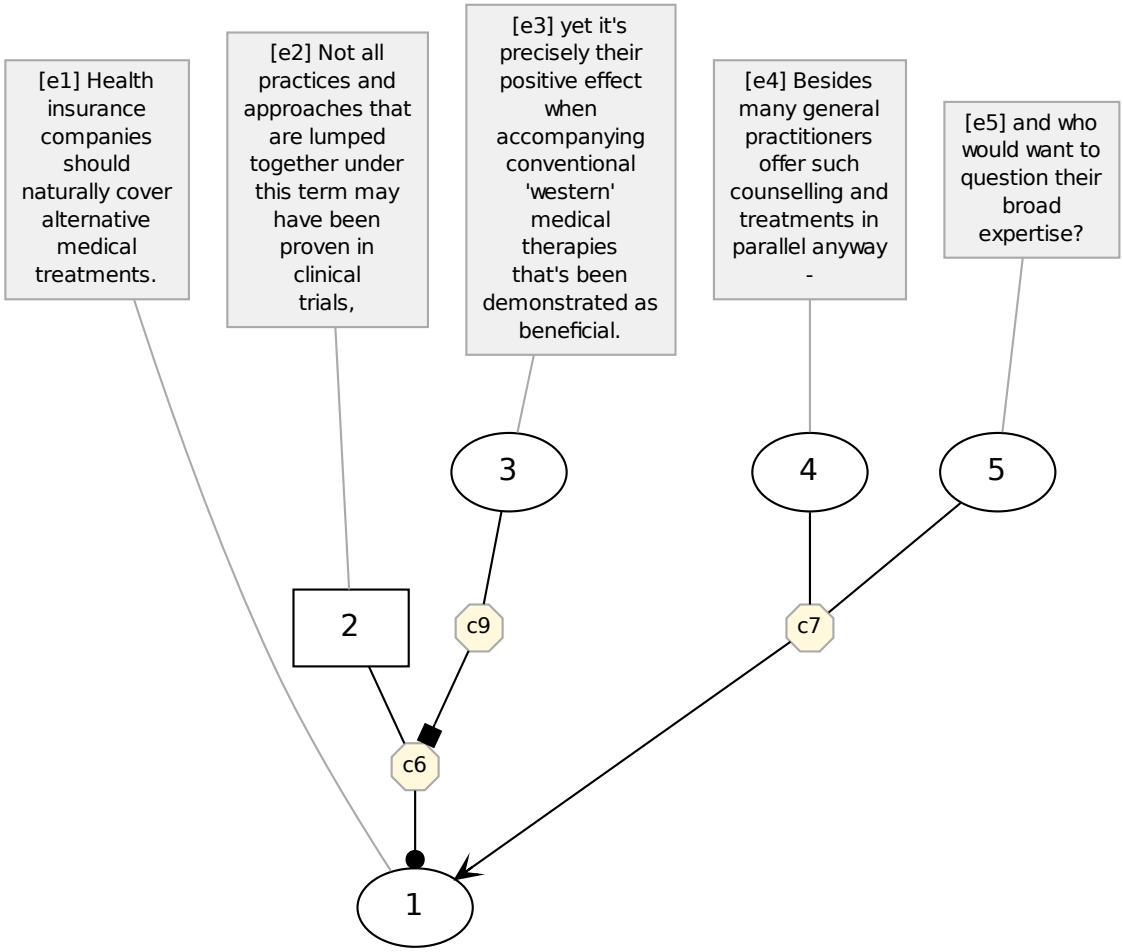


Figure 3.1: Argumentation structure of the example text

From a completely different background, Dung (1995) proposes a logical framework for simplified abstract argument structures, consisting only of claims and attacks. An *argumentation framework* (AF) is formally defined as a pair $(AR, attacks)$

where AR is a set of claims, and $attacks : AR \mapsto AR$ is a function describing which claims attacks which.[2]

Dung then defines various properties of AFs and associated lemmas, the first of which are:

- A set of claims $S \in AR$ is *conflict-free* iff its components don't attack each other, i.e.
$$\forall a, b \in S^2 \quad \neg attacks(a, b)$$
- A set of claims $S \in AR$ is *admissible* iff its components defend each other from attacks, i.e.
$$\forall a \in S, (\exists b \in AR\, attacks(b, a)) \implies (\exists c \in S\, attacks(c, b))$$
- Any AF has at least one *preferred extension*, a maximal (wrt set inclusion) admissible set of claims.

The rationale behind admissible sets of claims is that a debater who leaves counter-arguments unanswered is vulnerable, and admissible sets represents a well-rounded and believable argumentation. Dung introduces a high number of descriptive properties of argumentation frameworks, tying his formalism to applications in game theory, non-monotonic reasoning and logic programming. The structures remain abstract and quickly drift away from linguistic concerns.

3.3 Argumentation parsing

The literature dedicated to the automatic extraction of argumentative structures has long been sparse, even compared to the previous chapter's review of full-structure discourse parsing. However, in recent years, argumentation parsing has become a very active line of research. Initial works focus on solving specific related problems (identifying individual argumentative units, claims/premises, etc) without performing full parsing of the argumentative structure.

More recent works employ more sophisticated techniques involving global decoding over local probability distributions. This is something that follows similar trends from discourse parsing, an area with which argumentation parsing shares many commonalities, but has crucial differences too, as we will later see. The move towards more structured output prediction methods has only been natural since pipeline approaches suffer from error propagation problems.

[2]Hence the name. It's a very abstract framework.

Initial works. Moens et al. (2007) attempt to retrieve which sentences contain arguments, in a corpus of English texts from very diverse sources (news articles, legal proceedings, fora, speeches, etc). For this classification task, they employ both a naive Bayes method and a maximum entropy method, yielding an averaged 74% accuracy. Only shallow features are used for training, such as n-grams, verbs and adverbs present in the sentence; length of the sentence, average word length,[3] punctuation, and presence of keywords such as *but*, *consequently* or *because of*. Expanding on their work, Palau and Moens (2009) use the same techniques to predict whether a argumentative sentence is a *conclusion* (the core claim of an argument) or a *premise* (any other claim).

Florou et al. (2013) similarly identify arguments in Greek texts, using shallow features such as the presence of discourse markers and grammatical features of verbs (tense and mood).

Stab and Gurevych (2014b) performs a two-step decoding of local support structures, using two separate classifiers based on Support Vector Machines (Vapnik, 1995) and the corpus of argumentative essays described in Stab and Gurevych (2014a). Their first classifier identifies *argument components*, that is, whether a portion of a text corresponds to its central claim, to a secondary claim, to a supporting premise, or to a completely irrelevant utterance. Their second classifier identifies whether a pair of argumentative units is linked by a *support* relation, or not. Both models are trained with shallow features drawing from the basic structure of the text, lexical and syntactic analysis, as well as sets of PDTB discourse markers[4]

As we mentioned earlier, focused tasks in argumentation parsing have been performed by numerous works. However, as our efforts focus on the creation of complete argumentative structures of texts, we refer the reader to the very extensive documentation of Stab and Gurevych (2016) on argumentation tasks.

Full structures. Most approaches follow what Smith (2011) categorizes under polytope decoding, or decoding using specialized graph algorithms.

Peldszus and Stede (2015) learn local models which yield local probability distributions over ADUs and then perform global decoding using Maximum Spanning

[3]Quoting the publication: *"difficult" words might make the argument look more impressive.*

[4]*Penn Discourse TreeBank.* The feature sets used in discourse parsing (see section 2.2) and argumentative parsing often overlap.

Trees (MST). Their approach, which we expand on in our work, is further described in section 7.2.

Stab and Gurevych (2016) primarily work on their corpus of persuasive essays (Stab and Gurevych, 2014a), and also report results on the corpus of small texts used by Peldszus and Stede (2015). The authors of this *ArXiv* prepublication create a full parsing pipeline with the following steps:

- **Segmentation** of the source text, identifying the boundaries of argumentative units using a sequence prediction model, Conditional Random Fields (Lafferty et al., 2001);
- **Argument component identification**, using SVM as described previously in Stab and Gurevych (2014b);
- **Argumentative attachment identification**, also using SVM to detect whether arguments are linked or not (independently from the *attack* or *support* label);
- **Tree decoding**, using Integer Linear Programming to build a tree optimized with respect to the result of the two previous tasks;
- **Stance recognition** finally, classifying arguments by their voice, *proponent* or *opponent*.

Persing and Ng (2016) work on the very same corpus, with a very similar pipeline,[5] using the CoreNLP pipeline (Manning et al., 2014) for segmentation, maximum entropy classification for argument component and relation identification, and Integer Linear Programming for structure decoding, with an extensive set of constraints, among which:

- One *major claim* (with no parents) per essay, which must occur in the first or last paragraph;
- One parent per *premise*, which must be located in the same paragraph;
- The only parent of a *claim* must be the major claim;
- Each paragraph contains at least one claim or major claim;[6]
- Each sentence contains at most two components;
- Components never overlap on each other.

As we *also* employ similar techniques in our work, we discuss the ILP model of Persing and Ng (2016) in section 7.2.2 and perform a replication of their framework in section 7.2.3.

[5]Great minds think alike. Then again, the release of a new corpus such as Stab and Gurevych (2014a) probably suffices to explain the apparent synchronicity of new works

[6]Which is redundant with the combination of other constraints; otherwise some premises would have no claim to attach to.

Parallel work in discourse. As we mentioned earlier, argumentation and discourse parsing draw from the same techniques, be it their feature sets or their decoding mechanism.

Both MST and ILP techniques have been proposed as well in discourse parsing. In contrast though to discourse parsing, where different theories propose different underlying annotation schemes—trees for Rhetorical Structure Theory (RST)(Mann and Thompson, 1988), hyper-graphs transformed into dependency DAGs, for the Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003) or pairs of mostly adjacent sentences linked with implicit or explicit discourse markers for PDTB (Prasad et al., 2008)—underlying annotation schemes for argumentation use thus far only tree structures. Several relations, such as *undercut* can link arguments with other relations. Nonetheless, when transformed into dependency structures the end result is always a tree.

In discourse parsing, we demonstrated ourselves the use of ILP as a decoder for DAG structures (cf. section 6.2) while MST has been used repeatedly for tree structures (our own work in section 6.1, as well as Li et al. (2014)).

Chapter 4

The Settlers corpus

As described in section 2.1, existing corpora of annotated discourse either didn't provide full discourse structures, limiting annotations to pair of units (e.g. PDTB); or used tree-based representations of discourse (e.g. RST-DT). Additionally, few corpora studied multi-party dialogue, where incidentally tree-based structures aren't expressive enough.

The work on discourse parsing described in this thesis exploits the *Settlers* corpus (Asher et al., 2016), a corpus of multi-party chats annotated for discourse structure in the style of SDRT, based on chat logs of human players playing an online version of the game *The Settlers of Catan* (Teuber (1995); `www.catan.com`). The contents and purpose of the corpus are described in the following sections, as well as experiments highlighting some of its features.

4.1 *The Settlers of Catan*

Settlers is a win-lose multiplayer board game. 2 to 4 players compete to colonize the fictitious island of Catan. The process involves the acquisition of five kinds of resources (ore, wood, wheat, clay, sheep),[1] used to build roads, settlements and cities. Buildings earn players Victory Points (VPs); the first totaling 10 VPs wins the game. Buildings also allow players to receive new resources, according to the terrains surrounding them.

Every player's turn starts with a two-dice roll, which decides which terrains will produce resources for the neighboring settlements and cities. If a 7 is rolled, the player will instead move a special piece, the robber, which prevents the terrain where it's placed in from producing resources, and allows the player to steal a resource from an opponent.

[1]The official resource names are respectively ore, lumber, grain, clay and wool, which are usually ignored to reflect the icons of resource cards in the game.

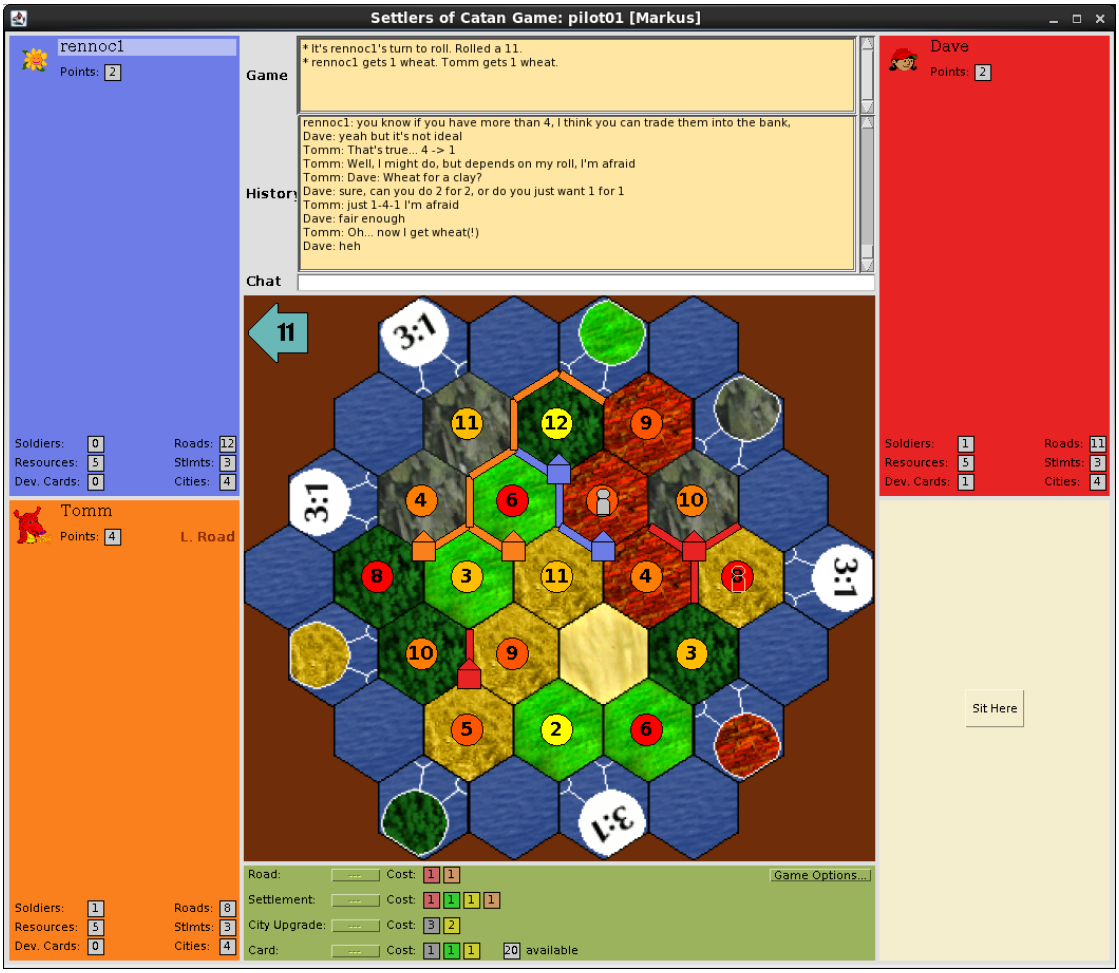


Figure 4.1: The JSettlers interface during a game.

A critical part of the game is the trade phase. Specific combinations of resources are needed to complete buildings: for example, a road requires one clay and one wood. In most games a player won't be able to gather all the needed resources from their personal production. One has several options to trade resources: with the game reserve, at a generally unfavorable rate; or with other players, through negotiations. Players converse to obtain what they want, and try to learn which resources their opponents need, or have. Resources stolen via the robber are kept secret, and player with more than 7 resources must discard half of them secretly when a 7 is rolled. As a result, players lack complete information about the possessions of their opponents.[2] Agents can, and frequently do, engage in 'futile'

[2]In most casual games of Settlers, players don't even bother to remember and track the resources exactly anyway.

negotiations that result in no trade (i.e., they miscalculate the equilibria).

Players in the *Settlers* corpus must chat in an online interface in order to negotiate trades, and each move in the chat interface is automatically aligned with the current game state—so one can compare what an utterance reveals about possessed resources with what the speaker actually possesses, and so identify examples of obfuscation (e.g., see table 4.1). The corpus consists of 59 games, each game containing from 100 to 900 dialogue turns, split into individual negotiation dialogues (up to several dozens per game).

Dialogue turn	Player	Utterance
157	gotwood4sheep	anyone got wood?
158	ljaybrad123	no
159	gotwood4sheep	ore for a wood, tomas?
160	tomas.kostan	yes but i need mine
161	gotwood4sheep	ore more?
162	tomas.kostan	2 ore for a wood?
163	gotwood4sheep	i don't have 2, sorry, just the one
164	gotwood4sheep	early doors, early offers :)
165	tomas.kostan	then i cannot make you a deal
166	tomas.kostan	sry
167	gotwood4sheep	ah dommage :(

Table 4.1: Excerpt from a dialogue.

4.2 What's so special about multi-party dialogue?

4.2.1 Non-tree-like structures

Multi-party dialogue or multi-party chat involves multiple participants who may address one or more others during their turn. For example, a person might ask a general question relevant to everyone present; once everybody has replied, that same person might reply to all of them with a single comment (e.g. thanking them) or with a single acknowledgment. Figure 4.2 provides such an example from our corpus. In turn 234, *gotwood4sheep* asks a question by making an underspecified offer to all other players. He then gets back negative responses to his question from *inca*, *CheshireCatGrin* and *dmm*; and then he broadcasts in 239 an acknowledgment of all the negative responses. That is, we have 235, 236 & 238 all attached to 234 as answers to the question in 234; and we have 239 that is attached to

235, 236 & 238 as an acknowledgment of the contents of those turns. The graph representation of the exchange is shown below the dialogue.

234	gotwood4sheep	anyone got wheat for a sheep?
235	inca	sorry, not me
236	CheshireCatGrin	nope. you seem to have lots of sheep!
237	gotwood4sheep	yup baaa
238	dmm	i think i'd rather hang on to my wheat i'm afraid
239	gotwood4sheep	kk I'll take my chances then...

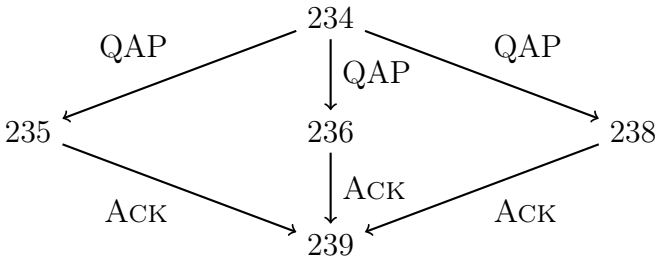


Figure 4.2: Dialogue excerpt showing the need for general graphs instead of trees.

The presence of such structures makes a powerful case that the general framework guiding the annotation of multi-party dialogues should take *non-tree-like* graphs as the underlying space of discourse structures. This requires a re-examination of the task of discourse parsing before attempting to learn such structures. In particular, the following questions arise:

- What are the common patterns found in non-tree-like structures? We expect dialogue to be coherent, which entails non-random behavior regarding structure. For instance, we expect units to be related more frequently to their neighbours, and rarely to distant units;

- If constraining discourse graphs to trees is too restrictive, what are the remaining constraints on discourse graphs? Are there hard constraints that all, or an overwhelming majority of structures, respect, so the can guide parsing in the same way tree-structure did in previous research?

- How far can traditional tree-based decoding mechanisms get us in dealing with such data? How well do they perform, and can we use them as a preliminary step for non-tree-like parsing?

4.2.2 Interwoven threads

Another complicated phenomenon in multi-party chat dialogues is the presence of crossing dependencies. Many theories of discourse structure like RST, given that they allow attachment only between adjacent spans, will not create structures with crossing dependencies. Also, theories that postulate a simple *right frontier constraint*, according to which only elements on the right frontier of a discourse structure (whether graph or tree) will not typically create structures with crossing dependencies. However, crossing dependencies are commonplace in multi-party chat. Several subgroups of interlocutors can and do momentarily form and carry on a discussion amongst themselves, forming thus multiple concurrent discussion threads. Since, though, what is being written is publicly available to all involved parties, it can be the case that participants of one thread might reply or comment to something said to another thread. Table 4.2 contains an example from our corpus, and figure 4.3 its associated structure.

165	lj	**anyone want sheep for clay?**
166	gw	**got none, sorry :(**
167	gw	so how do people know about the league?
168	wm	**no**
170	lj	i did the trials
174	tk	i know about it from my gf
175	gw	[yeah me too,]$_a$
		[*are you an Informatics student then, lj?*]$_b$
176	tk	did not do the trials
177	wm	**has anyone got wood for me?**
178	gw	[I did them]$_a$ [because a friend did]$_b$
179	gw	**lol william, you cad**
180	gw	**afraid not :(**
181	lj	*no, I'm about to start math*
182	tk	**sry no**
183	gw	**my single wood is precious**
184	wm	**what's a cad?**

Table 4.2: Example of interwoven threads.

There are at least three threads in this excerpt, highlighted with different fonts to aid the reader. The intuitive attachments in this excerpt involve the following crossing dependencies: (165, 168), (167, 170), (176, 178), (177, 179), (175, 181), (177, 182), and (180, 183). We note also the lack of standard discourse markers such as those found in the PDTB or RST manuals, "non-standard" orthography, the lack of elaborate syntactic structure and the frequent presence of sentence

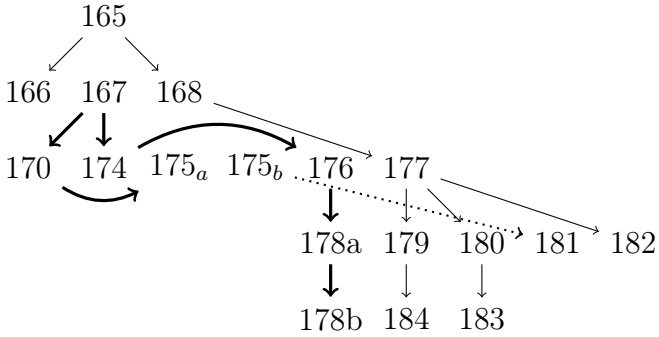


Figure 4.3: Structure of the interwoven threads of table 4.2.

fragments, all of which means we cannot rely on sentential syntax to aid with discourse parsing (syntax is very useful in monologue discourse parsing, as witnessed by the dramatically higher scores for intra-sentential discourse parsing (Joty et al., 2015)). Multi-party dialogue presents a discourse parsing problem free of syntactic crutches.

4.3 SDRT annotations

What *is* the *Settlers* corpus? We will give in this section a more practical overview of the dataset, with a description of the annotation model, as well as some useful statistics.[3]

The phenomena we just described are only part of the complications that appear in the discourse representation of multi-party dialogues, unfortunately rendering discourse theories based on attaching only adjacent units unsuitable for the annotation of the *Settlers* corpus. In order to be able to capture the discourse phenomena present in our chats, we decided to use Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2003). The theory not only allows long-distance attachments, which Ginzburg (2012) finds attested in multilogue, but also has semantics capable of dealing with fragments or non-sentential utterances (Schlangen, 2003), which are frequent in our corpus. Also, it can model non-tree like structures, which account for at least 9% of the links in our corpus (cf. section 4.2.1). Such structures make theories that model discourse structures with rooted trees, like *Rhetorical Structure Theory* (RST) (Mann and Thompson, 1988) or simple dialogue models where attachments are always made to *Last*—cf. (Schegloff, 2007; Poesio and Traum, 1997)—unsuitable.

[3]The corpus is available for download at `https://aclweb.org/anthology/attachments/D/D15/D15-1109.Attachment.zip`.

The raw data The entire corpus is based on the game log files of *JSettlers*[4], the open-source application enabling online play for *Settlers* , as well as a chat interface. 59 games have been recorded, out of which 36 had been fully annotated in the first release of the corpus. The log files contain very extensive information about the game events: dice rolls, building actions, and of course chat utterances. The latter are tied to the game state at their emission time, so we can track exactly the possessions of each player participating in the negotiations at any given point.

The log files, once parsed, give us the full chat history of the games. Given that bargaining sessions typically start after the dice roll opening each player turn, the history could be split into those sessions, one for each turn. However, discussions frequently span several consecutive turns, and the sessions were merged, accordingly, to mostly standalone dialogues. Whereas discussions in the midst of the game were fairly negotiation-focused (as expected), the discussions before and after the game often featured completely unrelated topics. As could be expected from chat logs, the text is messy from misspellings, contractions, missing punctuation and creative vocabulary.

The background structure The corpus possesses the following hierarchy, which has been manually annotated. At the top level, *dialogues*, ultimately treated as independent texts for the task of parsing. Inside them, *dialogue turns* are the consecutive utterances of a single player in the chat. While a dialogue turn may correspond to several log entries (the player sending their message in multiple parts), we grouped them together as they shared the same context. The final division of the text corresponds to *Elementary Discourse Units*, the arguments of discourse relations. In stark contrast to other corpora of discourse, a significant part of EDUs are one or two words long.

Each EDU has been associated with a *dialogue act*, representative of its role in the negotiations. The possible labels were:

- *offer*, initiating trade negotiations, even if they are often vague in practice, as in *anyone has clay?*
- *counter-offer*, a refinement of a prior offer, or a competing offer;
- *acceptance*, *refusal* and *other*, self-explanatory.

Cadilhac et al. (2013) creates a dialogue act prediction model for the *Settlers* corpus, which we reused as feature for our own parsing framework.

SDRT annotation process Annotation of the corpus involved four naive annotators and countless expert corrections of the structures,[5] involving five stages of

[4] `http://homepages.inf.ed.ac.uk/mguhe/socl/`

[5] Sometimes corrections of the data itself, as game log processing wasn't perfect, as well as segmentation tuning.

validation. In order to obtain meaningful results for our experiments, some games have been set apart to form a test set. The main statistics for the SDRT annotations are summarized in table 4.3. Please note that those statistics are extracted from the training and test corpora used in the experiments of chapter 6, and differ from the latest stats mentioned in Asher et al. (2016) by small amounts.

	Total	Training	Testing
Dialogues	1091	968	123
Turns	9160	8166	994
EDUs	10677	9545	1132
CDUs	1281	1132	152
Relation instances	10515	9423	1092

Table 4.3: Main statistics for SDRT annotations.

The corpus thus is quite sizable and has approximately the same number of EDUs and relations as the RST corpus (Carlson et al., 2003), the only other large corpus with full discourse structures for texts. Table 4.4 show the absolute frequencies of SDRT relation labels in the corpus. Three of the four most frequent labels in the corpus, QUESTION-ANSWER_PAIR, COMMENT and ACKNOWLEDGMENT witness the highly *reactive* nature of strategic multi-party chat.

Table 4.5 shows the distance between discourse relation arguments. Adjacent arguments correspond to a distance of one; if CDUs are involved, the CDU component which minimizes the distance is selected to compute it.

Table 4.6 shows the number of components of complex discourse units, which can be EDUs or CDUs (creating *recursive* CDUs).

	Total	Training	Testing
Question-answer pair	2546	2236	310
Comment	1869	1699	170
Continuation	984	870	114
Acknowledgment	947	839	108
Elaboration	874	776	98
Result	643	609	34
Q-Elab	592	528	64
Contrast	489	446	43
Explanation	430	397	33
Clarification question	239	220	19
Parallel	216	197	19
Correction	207	187	20
Alternation	153	134	19
Narration	134	120	14
Conditional	123	105	18
Background	63	60	3
TOTAL	10515	9423	1092

Table 4.4: SDRT label count of the corpus.

Distance	1	2	3	4	5	6	7	8-12	13+
Training	5889	1676	829	417	225	126	76	129	48
Test	697	220	94	31	15	10	4	12	4

Table 4.5: Distance between relation arguments.

Length	Total	Training	Testing
2	1121	982	139
3	137	119	18
4	13	12	1
5	7	6	1
6	2	2	0
7	1	0	1
TOTAL	1281	1121	160

Table 4.6: CDU count, by number of components

Chapter 5

Initial experiments: the Right frontier constraint and Extraction of hidden resources

This section describes two pieces of work exploiting the *Settlers* corpus, which are related to the parsing of discourse structure. The first subsection of this chapter presents an investigation of the Right Frontier Constraint and our efforts to expand it into multi-party dialogue, published in Hunter et al. (2015).

The second subsection presents a data extraction process, published in Perret et al. (2014), which is capable of extracting the resources that players of *Settlers* try to hide for strategic reasons. An artificial agent playing the game can leverage this probabilistic knowledge of its opponent's resources to improve its negotiation power.

5.1 A right frontier for multi-party chat

5.1.1 Importance of the RFC in multilogue parsing

In order to study the Right Frontier Constraint RFC for multi-party dialogues, we have chosen SDRT as our framework. As we have seen in the previous chapter, the *Settlers* corpus is already annotated for discourse structure in the style of SDRT. In addition, SDRT's RFC has been empirically validated on written monologue (newspaper articles and Wikipedia entries), using an annotation task in which annotators were not told about the RF, much less instructed to follow it (Afantenos and Asher, 2010). More importantly, however, SDRT deals easily with long distance attachments, which Ginzburg (2012) finds attested in multilogue, and has a semantics capable of dealing with fragments or non sentential utterances

(Schlangen, 2003), which are frequent in our corpus. Also, it can model non-tree like structures, like that shown in figure 5.1, which account for at least 9% of the links in our corpus. Such structures make theories that model discourse structures with rooted trees, like *Rhetorical Structure Theory* (RST) (Mann and Thompson, 1988) or simple dialogue models where attachments are always made to *Last*, cf. Schegloff (2007; Poesio and Traum (1997), unsuitable. In figure 5.1, QAP is the relation Question-Answer-Pair, ACK is Acknowledgement, and "kk" means "okay, cool".[1]

234	gw	anyone got wheat for a sheep?
235	inca	sorry, not me
236	ccg	nope. you seem to have lots of sheep!
238	dmm	i think i'd rather hang on to my wheat
239	gw	kk I'll take my chances then...

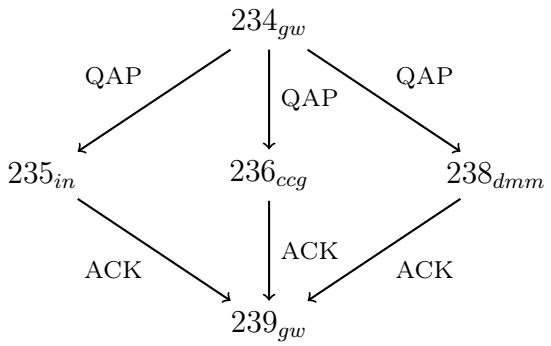


Figure 5.1: Example of a non-tree-like structure.

From the perspective of discourse processing, the RFC could be key in solving the *attachment problem*—that of predicting where a discourse unit π_n will attach to the structure for π_0–π_{n-1}. To put it in simpler terms, if there are no constraints at all concerning attachment, the search space of solutions is very large making thus attachment predictions impossible given the limited amount of data. So adding constraints is potentially interesting as it can limit the search space for a given approach. Of course, if attachment is already very constrained, adding an RFC makes little to no difference. In RST, attachment is restricted to adjunction over trees from contiguous spans, so the attachment problem is comparatively easy to solve.

SDRT is more liberal in its attachment principles than RST: though it incorporates constraints like connectedness, acyclicity and constraints on CDUs (Venant et al., 2013), non-adjacent and long distance attachments are common. Thus,

[1]For the clarity of examples, we will skip dialogue turns irrelevant to our main point.

adding an RFC to SDRT in principle greatly reduces the search space for attachment. When we combine this with the fact that SDRT's graphs can deal with examples like figure 5.1 and the examples of multiple threads discussed below, using SDRT to develop an RFC for multilogue is a natural choice.

5.1.2 Modifying the RFC

We recall the formal definition of the right frontier from section 2.3.2:

Definition 1 *For monologue, a node π_x is on the* RF *of a graph G, i.e.* $\text{RF}_G(\pi_x)$*, if either π_x is Last, or π_x is related to Last via a series of subordinating (Sub) edges, or π_x is a* CDU *that includes a node in* RF_G*. Formally, let $G = (V, E_1, E_2, Last)$ be a discourse graph.*

$$\begin{aligned}\forall \pi_x, \pi_y, \pi_z \in V \quad \text{RF}_G(\pi_x) \iff\ & \pi_x = Last \\ & \vee (\text{RF}_G(\pi_y) \wedge \exists e \in E_1, e(\pi_x, \pi_y) \wedge Sub(e)) \\ & \vee (\text{RF}_G(\pi_y) \wedge \exists e \in E_2, e(\pi_x, \pi_y))\end{aligned}$$

First modifications

SDRT's RFC relies on an incremental construction procedure that ensures that each EDU π_n is attached at some point along the RF of a connected graph G for EDUs $\pi_1, ..., \pi_{n-1}$ before π_{n+1} is even considered. Before developing an RFC for multilogue, we first need to modify this procedure to handle CDUs and backwards links. This subsection treats these topics in turn.

The incremental construction procedure assumes that it is possible to tell where a CDU will attach to an incoming discourse structure even before the full content of the CDU is known. Given that a CDU is a group of DUs that function together to form a single argument to a discourse relation, the incremental procedure potentially introduces a fair amount of guesswork into the process of reasoning about attachment. Consider (5.1) and the two possible continuations, (a) and (b).

(5.1) Bill: I'm running late$_{\pi_0}$ because my car broke down$_{\pi_1}$.
Janet: If you call Mike$_{\pi_2}$, ...

a. he might be able to pick you up and get you to the party on time$_{\pi_3}$.

b. he might be able to come over and fix your car$_{\pi'_3}$.

In (5.1a), $\pi_2 + \pi_3$ intuitively attaches to π_0, while (5.1b) suggests an attachment of $\pi_2 + \pi'_3$ to π_1. Until Janet utters the consequent, we can't tell where she is going with the antecedent.

There are two solutions to the problem posed by CDUs without resorting to a probabilistic version (which does not seem automatically learnable): (i) allow graphs to be corrected/repaired in light of new information (Asher, 1993; Prévot and Vieu, 2008) or (ii) wait to attach CDUs to an incoming discourse until the content of the CDU is complete. As an illustration, consider the graph G, shown in figure 5.2. We can, as shown in (i), construct G by first drawing an edge e_1 from π_x to π_y and then adding an edge e_2 from π_y to π_z and correcting e_1 so that its endpoint is the CDU (π_y+π_z). Alternatively, as shown in (ii), we can wait to draw an edge with π_x as initial point until the CDU (π_y+π_z) has been constructed. Relevant steps are separated by commas.

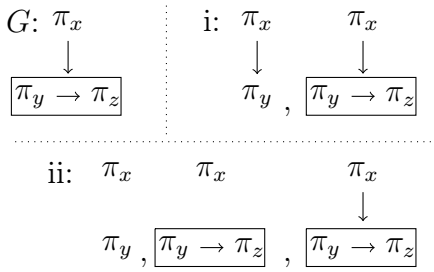
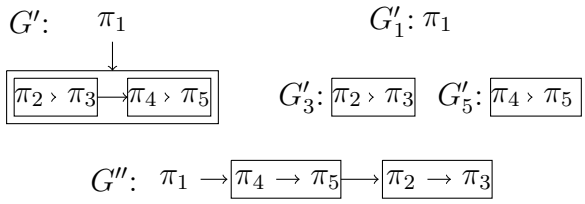


Figure 5.2: Corrected vs. delayed CDU construction

We adopt option (ii) and recast the RFC as a constraint on attaching sub-graphs. This makes the construction of an SDRS more compositional and allows us to reconcile the RFC with standard, non-incremental discourse parsing models. Even the standard case of EDU attachment can be thought of in this way. Let π_5 be an EDU that needs to be attached to a connected discourse graph $G_1 = \langle\{\pi_1, \pi_2, \pi_3, \pi_4\}, E_1, E_2, \pi_4\rangle$ and treat π_5 as the sole node in a graph $G_2 = \langle\{\pi_5\}, \emptyset, \emptyset, \pi_5\rangle$. The problem of attachment for π_5 can be recast as the problem of attaching G_2 to G_1.

To verify that a graph G contains no RF violations, we must be able to check for any sub-graph of G, whether that sub-graph violates the constraint. And we must allow that a sub-graph of G might contain further, unconnected sub-graphs, $G_1, G_2, ...G_n$, each with its own *Last*. Let G be an SDRS over EDUs $\{\pi_1, \ldots \pi_j, \pi_{j+1}, \ldots \pi_k, \pi_{k+1}, \ldots \pi_n\}$ and suppose we have constructed three sub-graphs G_j restricted to $\pi_1, \ldots \pi_j$ in their textual order, G_k restricted to $\pi_{j+1}, \ldots \pi_k$ in their textual order, and G_n restricted to $\pi_{k+1}, \ldots \pi_n$ in their textual order. G_j, G_k, and G_n each has its own RF, open to attachment, which makes possible highly undesirable graphs. Consider G' below and its sub-graphs G'_1, G'_3, and G'_5:
If we allow any sub-graph to attach to the RF of any other sub-graph, we could in theory, combine the sub-graphs of G' to build a graph G'' as follows:
In fact, every EDU in any graph G could be considered a single-node sub-graph, in which case allowing attachment on the RF of any graph would render an RFC



pointless. An utterance could serve as reaction to an arbitrarily later utterance, and speakers would be able to respond to points that haven't been salient for some time.

G'' is problematic because the CDU $\pi_2 + \pi_3$ is attached to $\pi_4 + \pi_5$, but neither π_4, π_5, nor $\pi_4 + \pi_5$ belong to the RF for π_2. Moreover, the RF for a new EDU, π_6, would be defined by π_5 (*Last* in G''), despite the the coordinating link from $\pi_4 + \pi_5$ to $\pi_2 + \pi_3$, which should block attachment to π_5.

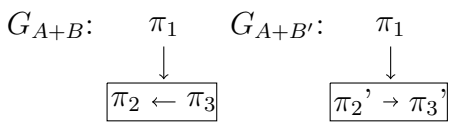
We need to constrain graph development. Let's return to our sub-graphs G_j, G_k, and G_n of G, and let G_{jn} be the extension of G_j with G_n. We must eventually construct a graph that attaches G_k to G_{jn}; call it $G_{jn}+G_k$. Such configurations can occur when G_k contains a parenthetical remark about G_{jn} or when it provides the topic. This means that G_k will be subordinate to G_{jn} or that $\text{RF}_{G_k} \cap \text{RF}_{G_{jn}+G_k} \neq \emptyset$. Let $\text{RFC}(G_{jn})$ mean that each edge in G_{jn} complies with the RFC in that each node π_n in G_{jn} attaches to a node on the RF for π_n as defined in Definition 1.

Another complication, given that edges in E_1 are directed, is that the direction of some edges reverses the textual order of their arguments.

(5.2) A [Would anyone give me some clay?]$_{\pi_1}$

B [I would,]$_{\pi_2}$ [if you give me a sheep]$_{\pi_3}$

B' [if you give me a sheep]$_{\pi'_2}$ [I would,]$_{\pi'_3}$



A+B yields a coherent SDRS, yet the backwards link $\pi_2 \leftarrow \pi_3$ violates the RF defined by Definition 1. The EDU π_1 is *Last* from the point of view of π_2, and so defines the RF for π_2; π_3 will not figure in this RF, thus the edge from π_3 to π_2 is a violation.

Furthermore, while (5.2B) is truth conditionally equivalent to (5.2B'), they are not discourse equivalent because $(\pi_2 + \pi_3)$ and $(\pi'_2 + \pi'_3)$ do not have the same felicitous continuations; i.e., $(\pi_x \rightarrow \pi_y)$ and $(\pi_y \leftarrow \pi_x)$ make importantly different contributions to discourse structure.

(5.3) [I would,]$_{\pi_2}$ [if you give me a sheep.]$_{\pi_3}$

a. [and an ore]$_{\pi_4}$

b. ??[with pleasure.]$_{\pi'_4}$

(5.4) [if you give me a sheep]$_{\pi'_2}$ [I would.]$_{\pi'_3}$

a. ??[and an ore]$_{\pi_4}$

b. [with pleasure]$_{\pi'_4}$

The examples above are noticeably more felicitous if the continuation targets the textually last EDU (π_3 or π'_3) despite the fact that these EDUs are the inputs for their respective conditional links.

To handle backwards links, we permit two graphs G_n and G_m to be attached with an edge in either direction.

We can now handle examples (5.3)-(5.4). Consider (5.4). In constructing the graph for (5.4a), π'_2 and π'_3 potentially determine separate sub-graphs. Suppose we attach π_4 to π'_2 to build the structure $[\pi'_2 \rightarrow \pi_4] \rightarrow \pi_{3'}$ (a felicitous combination of the EDUs in (5.4a)). $\pi_{3'}$ is the only node on the RF in the sub-graph consisting only of $\pi_{3'}$, so by Definition 1, it should remain on the RF once we attach it to $\pi'_2 + \pi_4$, but this will not be the case, as the RF will be defined by π_4, the *Last* node. Hence we predict that (5.4a) is unacceptable while (5.4b) is acceptable. Reversing the links makes no difference; while the highest link is reversed in (5.3), *Last* is determined by textual order, so *Last* is π_3 not π_2. Thus we cannot attach π'_4 to π_2 in (5.3b) for the same reason that we cannot attach π_4 to π'_2 in (5.4a).

5.1.3 Extending the modified RFC to multi-party dialogue

Our undirected RFC cannot yet handle structures like that in figure 5.1 (as neither 235 nor 236 are on the RF for 239) or examples of "interwoven threads", in which speakers juggle multiple conversations simultaneously. Both types of example are common in our corpus; the example in figure 5.3 involves (at least) three interwoven threads.

To handle such examples, we assign each speaker s in a multi-party dialogue a textual *Last*, i.e. the textually last EDU that s introduced into the chat. We call the RFC defined by allowing attachment to the *Last* of any speaker RFC+MLAST. RFC+MLAST allows the discourse parser to attach turns 235, 236 and 238 in figure 1 to turn 239 without violations, because for every edge with 239 as its endpoint, its initial point is *Last* for some speaker. For figure 5.3, MLAST lets 168 (*no*) attach to 165 as an answer, even though GW has introduced a separate question on a completely different topic that attaches via a coordinating Continuation relation

165	LJ	**anyone want sheep for clay?**
166	GW	**got none, sorry :(**
167	GW	so how do people know about the league?
168	WM	**no**
170	LJ	i did the trials
174	TK	i know about it from my gf
175	GW	[yeah me too,]$_a$
		[*are you an Informatics student then,* LJ*?*]$_b$
176	TK	did not do the trials
177	WM	**has anyone got wood for me?**
178	GW	[I did them]$_a$ [because a friend did]$_b$
179	GW	**lol william, you cad**
180	GW	**afraid not :(**
181	LJ	*no, I'm about to start math*
182	TK	**sry no**
183	GW	**my single wood is precious**
184	WM	**what's a cad?**

Figure 5.3: Example of interwoven threads.

to 165. Similarly, MLAST allows us to attach 175b to LJ's turn in 170 and GW's in 178 to 176 in spite of WM's attempt to start a new bargaining session. Likewise for the attachment of 182 to 177. RFC+MLAST fails, however, to allow the intuitive attachment of 181 to 175b, because GW's *Last* is 180 not 175b (cf. section 5.1.5 for discussion). Still, it yields considerable improvement over the modified RFC. Table 5.1 shows the effect of MLAST on RFC violations on the development portion of the *Settlers* corpus. The manually annotated structures obey RFC+MLAST on 95% of the links, while only 83.5% of the links obey the modified RFC.

5.1.4 Experiments and results for MLAST

A dynamic calculation of restrictions to the search space for attachments using basic RFC and RFC+MLAST shows that RFC+MLAST has a positive effect on the search space for dialogue parsing in the *Settlers* corpus. As shown in figure 5.4, the number of possible attachment points decreases dramatically with RFC+MLAST as the size of the dialogues in the corpus increases.

Using RFC+MLAST can have an important and beneficial effect on parsing. Yet just as the value of adding an RFC can vary depending on the discourse theory in question, it can also vary depending on the discourse parser in question. We have developed and trained learner and decoder dialogue parsers for attachment

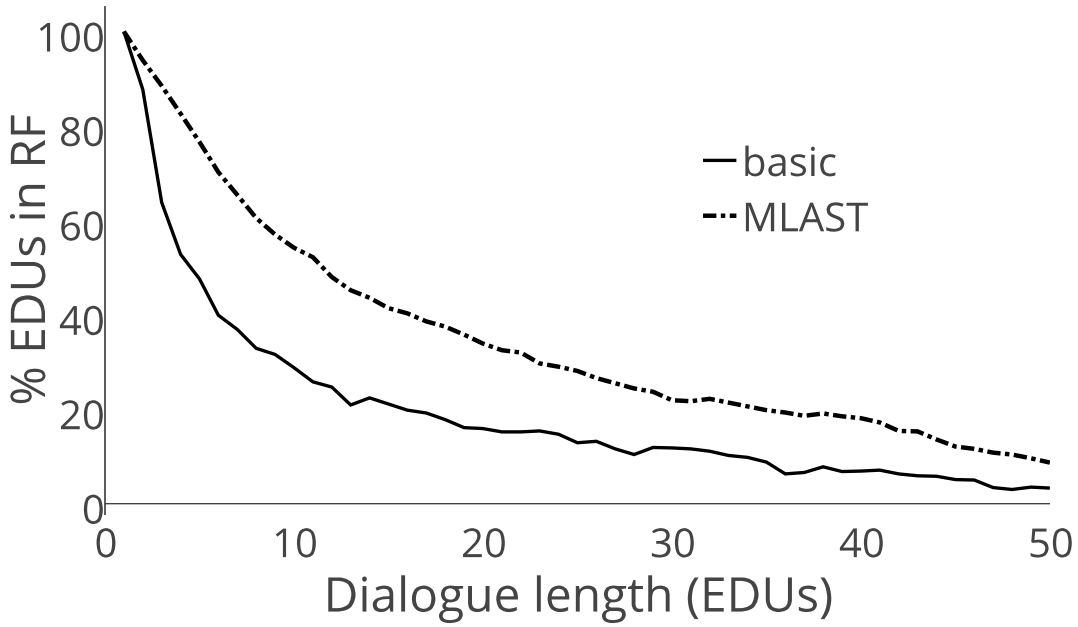


Figure 5.4: BASIC and MLAST versions of RFC

on a simplified version of the *Settlers* chat corpus (without CDUs). The learner is a regularized maximum entropy model (Berger et al., 1996). Using standard, superficial features for discourse parsing of the sort found in e.g., Muller et al. (2012b) and Li et al. (2014), we learn a probability distribution over pairs of EDUs as an input to several decoders. One decoder uses the MST algorithm (Chu and Liu, 1965; Edmonds, 1967). Another constructs first a maximal spanning directed acyclic graph, or MSDAG (McDonald and Pereira, 2006; Schluter, 2014) and then prunes it with constraints defined using ILP. The attachment F-scores for MST and ILP[2] without the RFC are provided in table 5.1.

Table 5.1 shows that MST closely complies with the standard RFC; 96,7% of its predicted attachments obey the RFC while 97,7% comply with RFC+MLAST. Therefore, using RFC+MLAST as a filtering constraint on MST would have little effect. ILP on the other hand could benefit considerably from having RFC+MLAST as a constraint, gaining up to 10% in its attachment score.

The data on MST, however, raise questions about its value as a parsing algorithm for our corpus. Note how closely it complies with the RFC. This is surprising, because CDUs are important in calculating the RF in both monologue and multilogue, so we would expect a considerable amount of RFC violations with a decoder that ignores CDUs. This is especially so given that removing CDUs from the gold

[2] *Integer Linear Programming*, a linear optimization technique we will encounter again in chapter 6.

Data	total links	RFC	MLAST	F-attachment
gold	9293	1536	447	100%
MST	8179	267	191	60.4%
ILP	17430	4342	2693	49.3%
LAST	8179	0	0	56%

Table 5.1: RFC violations

annotations on the *Settlers* corpus results in about a 10% increase in violations of the basic RFC; only 73% of the attachments in the manually annotated corpus obey RFC once we drop CDUs.

Let us consider the baseline, which we name LAST, where we simply attach each EDU to the preceding one. LAST verifies the plain RFC at 100%. The RFC violations over our corpus suggest that MST is much closer to LAST than it is to the gold annotations. The figures suggest that tree construction algorithms such as MST miss around 12% of the attachments in the gold corpus that are RFC violations but not violations on RFC + MLAST. Thus while MST might be a locally good strategy (with attachment F-scores at 0.81 within a sequence of consecutive turns by the same speaker), it is a globally mediocre strategy. This worsening echoes the difference reported by others between intra-sentential attachment scores and inter-sentential attachment scores in monologue (Joty et al., 2015). ILP, on the other hand, patterns more closely with the gold data and has many more long distance links.

5.1.5 Beyond MLAST

Double-tasking Recall that RFC+MLAST blocks the attachment of 181 to 175b in figure 5.3, because GW's *Last* is 180, and not 175b. This violation is interesting, because it illustrates a systematic pattern in which the same speaker carries on several interwoven threads, while others are talking. Such cases intuitively call for multiple *Last*s for a single speaker; that is, a *Last* for speaker s for each thread in which s is engaged. This notion, in turn, calls for a criterion for distinguishing threads.

One possible, and simple, solution would be to individuate threads by their members. Then we could extend the RFC+MLAST to include a *Last* for each speaker for each subset of speakers that is engaged in a thread. This would solve the problem of attachment in figure 5.3; however, it would not solve the problem in general, as we also have examples of multiple threads involving the very same subset of speakers. In figure 5.5, LJ and GW are engaged in both a trade negotiation, which takes place over turns 123-125, 127-129 and 131, and a thread about whether GW took logic, which takes place over turns 119, 126 and 130. Even if we

119	LJ	GW **did you take logic1 this year?**
123	GW	anyone got more clay? I fancy another
124	GW	can offer a range of items
125	LJ	i have clay
126	GW	**no i didn't LJ, I'm not a student :)**
128	LJ	would like wood
129	GW	1 for 1?
130	LJ	**ahhh ok, never mind**
131	LJ	sure

Figure 5.5: More interwoven threads in dialogue.

add a *Last* for each subgroup of speakers, 126, 128, and 130 will still give rise to RF violations.

It is difficult to define a thread precisely. And in fact, it's not clear to us that 126, 128, and 130 shouldn't count as RFC violations, in the same way that "discourse subordinations" (Asher, 1993) in monologue text count as RFC violations. Violations involving multiple threads with the same two speakers can be coherent but they require more effort to understand. For instance, annotators and interpreters could argue about the attachment of 130 to 126; and if we imagine that GW had made a different offer in 129 (say, 2 for 2 or 2 for 1), the we could easily imagine 130 as a response to 129. Moreover, GW actually refers to LJ by name in 126. This is a funny thing to do given that LJ is his only interlocutor at this point; if we treat 126 as an example of discourse subordination, however, then we can imagine that the name is being used as a signal for a discourse subordination.

Turn internal violations While we have not found a significant number of such examples in our corpus, the RFC might ultimately need loosening to handle examples like the following.

(5.5) B: Who has ore? **I have sheep to give**. *I could also give some clay.*
A': How many sheep?
B': ?? Three sheep even.

(5.6) A: Anyone want ore for sheep?
B: **I'm not giving up my sheep for now**, *but lj might want to give some of hers.*
A': What if I offer you two ore?
B': ?? Not for all the ore in the world.

Attachment possibilities for speakers are asymmetric. In (5.5)-(5.6), the boldface argument is related to the italicized argument by a coordinating relation (Alter-

nation in (5.5), Contrast in (5.6)), which should block the accessibility of the boldface argument. Indeed, B cannot continue with a comment targeting this argument (B+B'), though B' *would have been* felicitous in the absence of the italicized argument. By contrast, if another speaker, A, responds to B's turn, both arguments of the coordinating relation are accessible, as shown by the felicity of the A' continuations (B+A').

The theoretical explanation of this has to do with the underlying semantics of contributions in multilogue. The meaning of a dialogue is a set of commitment slates, one for each speaker. Speakers commit to their own contributions in dialogue but not necessarily to the contributions of their interlocutors, *unless* the attachments they make of *their own contributions* requires also that they take on board the commitments of the interlocutor (Hamblin, 1987; Lascarides and Asher, 2009). From this point of view, an asymmetry in the RFC is to be expected in multilogue.

5.2 Revealing resources

As suggested in the introduction, humans naturally extract meaning from discourse. We must not forget that this process has a purpose: for example, updating our beliefs about the world; communicate information efficiently; or in our chosen domain, bargain efficiently and win a board game. By a similar process, we must strive to leverage the information uncovered by discourse parsing.

To this effect, we propose in this section a practical application of discourse parsing: a framework for the task of extracting strategic information from online chat.

5.2.1 Motivation

When resources are limited, there is a fine line between agents cooperating and competing with one another for those resources, especially in a win-lose game. The goal of every rational agent is to maximize his *expected utilities* by finding *equilibrium strategies*: that is, an action sequence for each player that is optimal in that no player would unilaterally deviate from his action sequence, assuming that all the other players perform the actions specified for them (Yoam Sholam and Kevin Leyton-Brown, 2009). Calculating equilibrium strategies thus involves reasoning about what's optimal for the other players, which in turn depends on which resources they possess and which resources they need. However, almost every kind of bargaining game occurs in a context of imperfect information (Osborne and Rubinstein, 1994), where the opponent's current resources are hidden or non-observable.

Indeed, imperfect information often results from deliberate obfuscation: if an opponent can accurately identify your resources then they can exploit it for their own strategic advantage. For instance, in *The Settlers of Catan* (or *Settlers*), our chosen domain of investigation here, Guhe and Lascarides (2014) develop a *Settlers* playing agent where game simulations show that making the agent omniscient about his opponents' resources enables him to achieve more successful negotiations (i.e., a significantly higher proportion of his trade offers are accepted) and a significantly higher win rate than his non-omniscient counterparts. So it is rational for players to balance achieving their desired trades with revealing as little as possible about their own resources, while at the same time attempting to elicit information about their opponents' resources.

In negotiations using natural language dialogue, eliciting information about an opponent's resources is often realized as a question; the opponent, on realizing the question's purpose, often avoids revealing their resources in their response. They use various communicative strategies to achieve this effect, such as making a counteroffer, being vague, or simply changing the subject.

5.2.2 Annotation of the training data

Base corpus. Our model is trained on the *Settlers* corpus, described in chapter 4. As a reminder, in a game of *Settlers*, players acquire resources (ore, wood, wheat, clay, sheep) to build roads, settlements and cities, through dice rolls. They can also acquire missing resources through trading with other players—so players converse to negotiate trades. Players lack complete information about their opponents' resources. Consequently, agents can, and frequently do, engage in 'futile' negotiations that result in no trade (i.e., they miscalculate the equilibria).

Players in the corpus described in Afantenos et al. (2012) must chat in an online interface in order to negotiate trades, and each move in the chat interface is automatically aligned with the current game state—so one can compare what an utterance reveals about possessed resources with what the speaker actually possesses, and so identify examples of obfuscation (e.g., see table 5.2). The corpus consists of 59 games, and each game contains dozens of individual negotiation dialogues, each dialogue consisting of anywhere from 1 to over 30 dialogue turns. In our experiments, we have used 7 games consisting of more than 2000 dialogue turns (see section 5.2.2).

Table 5.2 contains an excerpt from one of the dialogues. In turn 157 the player GW asks if anyone has any wood, implying that he wants to negotiate an exchange of resources where he receives wood. Player LJ is the first to reply, negatively, implicating that he has no wood. Turn 158 is thus annotated with the information

Turn	Player	Utterance
157	GW	anyone got wood?
158	LJ	no
159	GW	ore for a wood, tomas?
160	TK	yes but i need mine
161	GW	ore more?
162	TK	2 ore for a wood?
163	GW	i don't have 2, sorry, just the one
164	GW	early doors, early offers :)
165	TK	then i cannot make you a deal
166	TK	sry
167	GW	ah dommage :(

Table 5.2: Excerpt from a dialogue.

that the player LJ is revealing that he has 0 wood.[3] In turn 159 player GW persists in his attempt to negotiate, referring directly to player TK and making a more specific trade offer, of ore in exchange for wood. He has thus revealed that he possesses at least one ore. The player TK acknowledges that he has wood (so this turn is annotated with the information that TK has at least one wood) but that this resource is important to him. TK then proposes two ore in exchange for one wood (again, this turn is annotated with the information that TK possesses at least one wood). GW in turn 162 explicitly says that he has only one ore and not two, so this turn is annotated with the information that player GW has exactly 1 ore. In the end the negotiation fails since for TK a wood is currently worth more to him than what GW is currently offering.

Note that revealed resources depend not only on the content of the individual utterance but also on its semantic connection to the discourse context. For example, the dialogue turn 158 (*no*) reveals nothing about resources on its own; it is the fact that it is connected to the question 157 with a QAP (QUESTION-ANSWER-PAIR) relation that commits LJ to having zero wood. Similarly, 160 is an ACKNOWLEDGMENT to 159 and so reveals that TK possesses at least one wood.

Annotation process. We manually annotated each utterance with its corresponding revealed resources. Two annotators (including myself) were involved in the task. After a thorough examination of the dialogues in an initial game, we

[3]In this paper, we simplify our task by ignoring the fact that players can lie. As matter of fact, manual analysis of the corpus logs show that players rarely lie concerning their resources, preferring instead to conceal relevant information by avoiding giving a direct answer.

settled on the format of the annotations and the guide for performing the annotation task. The annotation format is as follows: each speech turn corresponding to a revealed resource is annotated with a pair: a resource name, and the quantity interval which the player reveals, representing the lower and upper bound of the resource. For example, in dialogue turn 158 of table 5.2 player LJ declares that he has no wood, so this dialogue turn is annotated as `(wood, [0,0])`. In dialogue turn 159 player GW reveals he has at least one ore, so this turn is annotated as `(ore, [1,+∞])`. Revelations of multiple resources are associated with multiple pairs.

To test the consistency and difficulty of the task, both annotators independently annotated a single game after settling on the above format and instructions for annotation. Over 422 speech turns, the resulting kappa coefficient of inter-annotator agreement is **0.94**, enough to validate our annotation method. The remaining 6 games were then annotated, for which statistics can be found in tables 5.3 and 5.4. Most dialogues appear to be short, frequently consisting of comments on the game status, which do not call for answers. Trade negotiations are usually longer, with players emitting offers and counteroffers, sometimes competitively. Revelations of resources are present in 21% of dialogue turns.

Speech turns	2201
Dialogues	263
Word count	9121
Turns revealing resources	452 (21% of turns)

Table 5.3: Overview of the annotated dataset.

Number of speech turns in dialogue	Dialogue count
1-5	112
6-10	63
11-15	23
16-20	13
21 and more	23

Table 5.4: Dialogue statistics of the annotated dataset.

5.2.3 Formulating the problem

As mentioned earlier, our goal is to predict whether a given turn reveals that its emitter possess a resource, and if so the type of the resource and its quantity in the form of an interval. Although players could potentially reveal having a specific number of resources (e.g., line 163 in table 5.2), in most cases the players reveal either having zero resources (interval `[0,0]`) or having at least one (interval `[1,`∞`]`), and in few occasions, players reveal that they have more than one (interval `[2,`∞`]`) or exactly two resources (`[2,2]`). In most of the cases, a revelation of having zero resources is manifested through the player rejecting a trade offer by stating that they don't have the resource desired by their opponent.

Using a single classifier to predict from an NL string the revelation of a particular type of resource, or no revelation of any resource, would involve classifying each utterance into 6 classes: one for each of the 5 types of resources, and one for revealing that no resources are possessed. But such a model would fail to take full advantage of the following facts. First, the NL strings that reveal a resource are relatively invariant, save for the particular resource type; in other words, the ways in which people talk about their possession of clay is the same as their talk about possessing wood, save for the words *clay* vs. *wood*. Secondly, it is easy to specify the properties of a revelation (both the type of resource and quantity) when we know a given utterance exhibits a revelation. Given these observations, we decided to divide the prediction process into two sub-tasks:

1. Determine if a given speech turn reveals a resource or not;
2. For those utterances that do reveal a possessed resource, determine the type of resource and its associated quantity interval.

5.2.4 Classification of revealing turns

Our goal is to learn a function

$$f : \mathcal{X} \mapsto \{0, 1\}$$

where every $\mathbf{x} \in \mathcal{X}$ corresponds to a vector representing a dialogue turn and $\{0, 1\}$ represents the fact that there is a revelation concerning an unspecified resource from the part of the dialogue act emitter.

Features. The (mostly shallow) features that we have extracted for every dialogue turn can be summarized in the following categories:

- **Contextual features**: positioning of the turn in the dialogue;

Category	Description
Contextual	Speaker initiated the dialogue
	First utterance of the speaker in the dialogue
	Position in dialogue
Lexical	Contains resource name
	Ends with exclamation mark
	Ends with interrogation mark
	Contains possessive pronouns
	Contains modal modifiers
	Contains question words
	Contains a player's name
	Contains emoticons
	First and last words
Pattern-related	Contains a possession structure, such as *I have (no) X*
	Contains a query structure, such as *I need X*
	Contains *X for Y*
Relational	Is predicted as question wrt another speech turn
	Is predicted as answer wrt another speech turn

Table 5.5: Feature set description.

- **Lexical features**: single words present in the utterance;
- **Pattern-related features**: recurring speech structures associated with revealed resources;
- **Relational features**: discourse relationships with other turns.

These features are listed more extensively in table 5.5. Non-relational features are extracted directly from the underlying text. In order to compute the relational features—essentially whether a pair of dialogue turns are linked with a *Question-answer pair* (QAP) or a *Question-Elaboration* (Q-Elab) discourse relation—we used the results of a separate classifier for the prediction of discourse relations. This classifier was trained on 7 games consisting of 2460 dialogue turns. We used a Maximum Entropy classifier, as in the case of predicting revealed resources (see below for more details). We selected, for this classifier, a subset of the feature set used for the task of predicting revealed resources. More specifically, we used only the *Contextual* and *Lexical* features shown in Table 5.5. Although the model we have used was a general one, capable of predicting the full set of SDRT discourse relations used in the *Settlers* corpus, for this series of experiments we were only

	Precision	Recall	F1 score
QUESTION-ANSWER PAIR	83.8	86.8	85.3
Q-ELAB	53.3	57.9	55.5

Table 5.6: Results for the relation prediction task.

interested in the QAP and Q-ELAB relations. Results for these relations are shown in table 5.6.

Probabilistic model. For our classifier, we used a regularized maximum entropy (MAXENT, for short) model (Berger et al., 1996). In MAXENT, the parameters of an exponential model of the following form are estimated:

$$P(b|t) = \frac{1}{Z(c)} \exp\left(\sum_{i=1}^{m} w_i f_i(t, c)\right)$$

where t represents the current dialogue turn and c the outcome (i.e., revelation of a resource or not). Each dialogue turn t is encoded as a vector of m indicator features f_i (see table 6.1 for more details). There is one weight/parameter w_i for each feature f_i that predicts its classification behavior. Finally, $Z(c)$ is a normalization factor over the different class labels (in this case just two, whether we have a revelation of a resource or not), which guarantees that the model outputs probabilities.

In MAXENT, the values for the different parameters $\hat{w}$ are obtained by maximizing the log-likelihood of the training data T with respect to the model (Berger et al., 1996):

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_{i}^{T} \log P(c^{(i)}|t^{(i)})$$

Various algorithms have been proposed for performing parameter estimation (see Malouf (2002) for a comparison). Here, we used the Limited Memory Variable Metric Algorithm implemented in the MegaM package.[4] We used the default regularization prior provided by MegaM.

5.2.5 Predicting the type and quantity of revealed resource

From our observations, the majority of utterances revealing resources fall into one the following two categories:

[4] Available from `http://www.cs.utah.edu/~hal/megam/`.

Type	Keywords
Negation	no, not, don't
Second-person	you, someone, anyone
Possession	got, have, give, spare, offer
Query	want, need, get
For	for

Table 5.7: Markers used in type prediction

- **Self-contained**: resource and quantity can be deduced from the utterance alone, such as *I have no ore*;

- **Contextual**: some information is deduced from another utterance. Both usually form a question-answer pair, such as *Do you have any wheat? – Yes.*

We created five marker categories, described in table 5.7, from the most frequent words appearing in revealing utterances. We designed a rule-based model using these markers; their combination allows us to pinpoint where the resource the player reveals is mentioned. For example, in the utterance *anyone has sheep for ore?*, the second-person marker *anyone* and the possession marker *has* indicate that the first mentioned resource is the one wanted by the player, which he doesn't reveal as possessing. Moreover, the presence of a *for* marker indicates that the players offers a resource. Hence, the resource following the marker, *ore*, is possessed by the player.

Such a rule system allows us to analyze a single utterance. However, in the case of a QAP, we often fail to retrieve data from the answer utterance alone. A second pass is thus performed on the question utterance, giving us enough context to deduce revealed resources. For example, in the QAP *anyone have wood? – none, sorry*, in the second utterance, the negation marker *none* implies the absence of an unknown resource. The processing of the first utterance reveals that *wood* is requested by another player. We conclude that the answering players possess no wood.

We first tested our rule model on reference data, knowing exactly (from the annotations) which speech turns contained revealed resources, and which discourse relations linked them. We then used the model on predicted data (discourse relations as well as dialogue turns representing revealed resources), effectively creating a full end-to-end system.

Baseline (accuracy : 82.1)			
	Precision	Recall	F1 score
H_+	54.7	73.7	.628
H_-	92.5	84.2	.882
Our method (accuracy : 89.2)			
	Precision	Recall	F1 score
H_+	75.2	70.6	.728
H_-	95.2	94.0	.933

Table 5.8: Results for the task of detecting whether a turn reveals a resource. H_+ represents the hypothesis that the dialogue turn does reveal a resource, while H_- the hypothesis that it doesn't.

5.2.6 Experiments and results

The classifier is trained using 10-fold cross-validation. For every training round, we partition the data by dialogues. 90% of them (resp. 10%) are then used to train (resp. evaluate) our model. We compared our method to a baseline, which does not involve machine learning. This naive model predicts revealed resource whenever a resource is mentioned by name in the utterance.

After performing ten rounds of cross-validation on the training data, we achieve a F1 score of **0.72** for the positive hypothesis *"This speech turn reveals a resource"*. The opposite class ("There is no revealed resource in this turn") has an F1 score of 0.93, achieving thus a global accuracy of 89.2%. Detailed results for our model and baseline are shown in table 5.8.

Results for the prediction of resource type quantity interval are shown in table 5.9. As we can see, prediction of the type of resource that a player's dialogue turn reveals has an accuracy of 77% on the manually annotated instances, which falls down to 61.5% when using the results of the first classifier as input. Interval prediction on the other hand has an accuracy of 79.9% when using manually annotated results which falls down to 65.7% when using the results of the first classifier as input. Note as well that we have implemented a baseline for both systems. Concerning resource type, the baseline randomly attributes a resource to utterances labeled as revealing one. The baseline for interval prediction assigns the most frequent interval. Results are also shown in table 5.9.

In table 5.10 we report results on the pipeline combining the three tasks. The accuracy of 57.1% does not include the instances that have been classified as not revealing any resources by the first classifier. When we evaluate both classes the accuracy goes up to 86.3%.

Accuracy	on manual annotations	on the output of the first classifier
	Baseline (random)	
Resource type	0.165	0.146
Interval	0.559	0.328
	Our method	
Resource type	0.770	0.615
Interval	0.799	0.657

Table 5.9: Baseline and evaluation of predicting resource type and interval.

	Accuracy
On all instances	0.863
Only on instances classified as revealing a resource	0.571

Table 5.10: Results of the pipeline, that is prediction of the exact triplets `(resource, [lower bound, upper bound])`.

Discussion

The first step of our prediction process, locating turns revealing resources, yields very encouraging results (see Table 5.8): we are able to retrieve such turns with an F1 score of over 0.72, while they represent only 21% of all speech turns. On the other hand our system does not perform very well on the detection of resource type as well as the associated interval. This is to be expected: since we have split our system in three parts, there is error propagation in the pipeline. On the other hand jointly predicting the triplets is not a viable solution either, since this would lead to a great number of classes (six as we have mentioned above, multiplied by all the possible values for lower and upper bounds). We would like though to note that we greatly outperform both baselines for each of the last two tasks.

One way to improve the quality of our prediction would be to add more relational features. As context plays a critical part in determining the meaning of an utterance, features associated to its relational neighbors should be taken into account. This is true for the prediction of whether a dialogue turn reveals a resource as well as for the prediction of its type.

Accuracy for this last task is not very satisfying. The main reasons for this, which can serve as the basis for future improvements, include:

- **Ambiguous *for* patterns.** The utterance `X` *for* `Y` can be interpreted two

ways : either as a revealing possession of `X` or `Y`. This is ambiguous even for the players themselves since often they pose a clarification question. Observation shows that the latter (possession of `Y`) is more frequent. The rule model implements this behavior as default when encountering such a pattern. In actual dialogues, this ambiguity is resolved by a follow-up question (*Which one are you offering ?*) or by the game context (dice rolls and resource distribution) which we haven't access to.

- **Long-distance resource anaphora.** On most trade negotiations, the resource being traded isn't mentioned by name at every point of the discussion, but rather referred to implicitly. When this carries over several speech turns, it becomes increasingly difficult to determine the traded resource (solving the anaphora) from a later utterance. Incorporating anaphora resolution could definitively improve our results.

- **Uncommon idioms.** Some utterances, such as *I'm oreless*, or *I just discarded all of my sheep*, employ rare vocabulary (with respect to the corpus) to describe resource possession. Incorporating more lexical information is necessary.

Chapter 6

Parsing dialogue structure

In this chapter we present our work on predicting full discourse structures. Most previous work on discourse parsing has focused on monologues; to the best of our knowledge this thesis is the first work to fully study discourse parsing for multi-party dialogues. Our approach consists in performing global decoding over a local model which learns a probability distribution of attachments and relation types between pairs of EDUs. In addition, we provide mechanisms to remove CDUs from SDRT structures, converting into dependency graphs, which are more accessible to available parsing techniques.

Our first approach (Afantenos et al., 2015) explores decoding using Maximum Spanning Tree decoding on top of a probabilistic model of local discourse relations. We introduce a simple mechanism of CDU elimination.

Our second approach (Perret et al., 2016) performs decoding using Integer Linear Programming, which enables us to predict directed acyclic graphs (DAGs) instead of trees. We introduce and motivate a set of constraints for multi-party dialogues, as well as improved mechanisms of CDU elimination.

Formalism In this chapter we will heavily use the SDRT formalism presented in section 2.1.3. As a reminder, for a dialogue D segmented in n EDUs, i.e. $D = \{e_1, \ldots, e_n\}$, a SDRT structure is defined as a tuple (V, E_1, E_2, ℓ), where:

- $V = D \cup \Pi$ is a set of nodes or discourse units, with Π as the set of CDUs ;
- $E_1 \subseteq V \times V$ is a set of edges representing discourse relations;
- $E_2 \subseteq V \times \Pi$ is a set of edges that represents parthood in the sense that if $(x, y) \in E_2$, then the unit x is a component of the CDU y;
- $\ell \colon E_1 \to R$, is a labeling function that assigns an edge in E_1 its discourse relation type (R being the set of SDRT relation labels).

6.1 Tree decoding

6.1.1 Dependency structures

For a given discourse graph for SDRT of the form (V, E_1, E_2, ℓ), there is, as of today, no general and reliable method to calculate edges in E_2 (i.e. the CDUs); and no such method has been presented in the literature. In order to perform constrained decoding over local probability distributions, we have opted for a strategy first presented in Muller et al. (2012b) for SDRT. The strategy involves transforming hyper-graphs into dependency graphs. We transform our full graphs (V, E_1, E_2, ℓ) into dependency structures $(D, E'_1, \varnothing, \ell')$, D being the set of EDUs, by replacing any attachment to a CDU with an attachment to the CDU's head—the textually first EDU within the CDU which has no incoming links. Our transformation in effect sets E_2 in our general definition of a graph to $\varnothing$. In the case that we have a discourse relation between two EDUs, this relation is kept intact since it already represents a dependency arc. In case a discourse relation has one or two CDUs as arguments, the CDUs need to be replaced with their *recursive head*. In order to calculate the recursive head we identify all the DUs with no incoming links; if they are CDUs we recursively apply the algorithm until we get an EDU. If there is more than one EDU with no incoming links we pick the leftmost, i.e. the one firstly introduced in the text. Figure 6.1 shows an example of such a transformation.

Hirao et al. (2013) and Li et al. (2014) later followed a similar strategy for the creation of dependency structures for RST. Every single nucleus-satellite relation was transformed into a dependency relation with the governor being the EDU representing the nucleus and the dependent being the satellite. For relations between non-EDU higher spans, the recursive head was used. It is unclear how Li et al. (2014) deal with binary multi-nucleus relations like CONTRAST for example; it is not clear how to calculate the recursive head of the span.[1] In such cases an arbitrary decision—like always taking as the nucleus the leftmost or the rightmost span—has to be taken. In the SDRT annotations, however, every edge in the graph is already directed and so such arbitrary decisions can be avoided.

The above transformation gives us a directed acyclic graph $G = (D, A)$ for each dialogue D such that:

- $D = \{e_1, \ldots, e_n\}$;
- $A \subset D \times D \times R$, where R is the set of SDRT relation labels ;
- if $(e_i, e_j, r) \in A$ then $\forall r' \neq r$, $(e_i, e_j, r') \notin A$, ensuring that our graph is not a multi-graph (only one relation exists between two EDUs).

[1] Although Li et al. (2014) do explain how to treat n-ary multinuclear relations, following others (Hernault et al., 2010, for example).

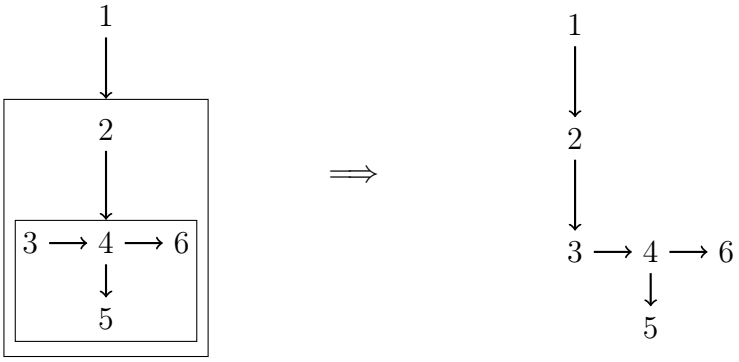


Figure 6.1: Translation of SDRT discourse graphs into dependency structures. In the left figure, the CDUs are displayed as boxes for greater clarity, as in section 5.1.

6.1.2 The turn constraint

Given our observations about the structure of dialogues in our corpus, we hypothesize that a dialogue is fundamentally sequential: first one person talks and then others react to them or ignore them, but the discourse links that do occur between speaker turns are *reactive*. In other words, a turn can't be anaphorically and rhetorically dependent on a turn that comes after it. Thus, the nature of dialogue imposes an essential and important constraint on the attachment process that is not present for monologue or single-authored text, where an EDU may be dependent upon any EDU, later in the ordering or not: in dialogue there are no "backwards" rhetorical links such that an EDU in turn n by speaker A is rhetorically and anaphorically dependent upon an EDU in turn $n + m$ of speaker B with $A \neq B$. We call this the *Turn Constraint*. Within a turn, however, just as in monologue (as is evident from a study of most styles of discourse annotations of text), backwards links are allowed.

Given this observation, we decided to split our local model into two different ones. The first one concerns the learning of a model for intra-turn utterances,[2] while the second models inter-turn utterances. The intra-turn model considers as input during learning all pairs of EDUs (i, j) with $i \neq j$. The inter-turn model on the other hand does not contain any backward links during learning. In other words it takes as input all pairs of EDUs (i, j) with $i < j$. We apply the turn constraint not only during learning of the local models, but also during decoding. This practice is also followed—at the sentence level—for monologues (Wellner and Pustejovsky, 2007; Joty et al., 2012; Joty et al., 2013), though our turn constraint,

[2]EDUs are considered as belonging to the same turn if they are by the same speaker without any interjection from an other speaker. In other words any consecutive EDU by the same speaker is considered as belonging to the same turn.

we believe, is firmly supported not only by our data but also by a good theoretical model of dialogue.

6.1.3 Local model of discourse relations

Ideally, we want to be able 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 features for a pair of EDUs and $\mathcal{Y}_R$ represents the set of SDRT relations. The upshot of this is that we are building a local sort of model that learns relations between individual EDUs with a certain probability but does not learn a global or even local structure.

Feature extraction To train our local models, we extracted features for every pair of EDUs in a given dialogue. Our features concern the pair of EDUs as well as features related to each EDU specifically.

For any given dialogue (as defined previously), every pair $(u_i, u_j) \in E^2$ of EDUs it contains will correspond to a feature vector $x_{ij} \in \mathcal{X}_{E^2}$, of the form

$$\begin{aligned} x_{ij} = (&p_1(u_i, u_j), ..., p_m(u_i, u_j), \\ &s_1(u_i), ..., s_n(u_i), \\ &s_1(u_j), ..., s_n(u_j)) \end{aligned}$$

so that each vector represents a set of *pair features* and two sets of *singular features* for every pair of EDUs.

The feature set, detailed in table 6.1, can be summarized as follows:

- **Positional features**: (related to) the non-linguistic context of the pair;
- **Lexical features**: single words[3] and punctuation present in the EDUs;
- **Parsing features**: syntactic dependency[4] and dialogue act[5] tagging.

[3]We use a number of lexicons (opinion markers, quantifiers, PDTB markers, etc.), each corresponding to a feature

[4]Provided by the Stanford CoreNLP pipeline (Manning et al., 2014).

[5]The prediction model of Cadilhac et al. (2013) generates EDU tags such as *Offer*, *Refusal*, etc.

Category	Description
Positional	Speaker initiated the dialogue
	First utterance of the speaker in the dialogue
	Position in dialogue
	Distance between EDU*s*
	EDU*s have the same speaker*
Lexical	Ends with exclamation mark
	Ends with interrogation mark
	Length in lemmas
	Contains possessive pronouns
	Contains modal modifiers
	Contains words in lexicons
	Contains question words
	Contains a player's name
	Contains emoticons
	First and last words
Parsing	Subject lemmas given by syntactic dependency parsing
	Predicted dialogue act

Table 6.1: Feature set description. Pair features are italicized.

Local probability distributions We use a regularized maximum entropy model (shortened as MaxEnt) (Berger et al., 1996). In MaxEnt, we estimate the parameters of an exponential model of the following form:

$$P(r|p) = \frac{1}{Z(c)} \exp\left(\sum_{i=1}^{m} w_i f_i(p, r)\right)$$

where p represents a pair of EDUs and r the learnt label (i.e. the type of relation, or a binary attachment value between the two EDUs). Each pair of EDUs p is encoded as a vector of m indicator features f_i (detailed in table 6.1). The parameters learned by the model are the weights w_i, associated to each feature f_i (a higher weight translating to a higher influence on the classification). Finally, $Z(c)$ is a normalization factor over the different class labels, which guarantees that the model outputs valid probabilities. In MaxEnt, the final values for the different parameters $\hat{w}$ are obtained by maximizing the log-likelihood of the training data T with respect to the model:

$$\hat{w} = \underset{w}{\operatorname{argmax}} \sum_{i}^{T} \log P(r^{(i)}|p^{(i)})$$

Various algorithms have been proposed for performing parameter estimation (see (Malouf, 2002) for a comparison). In this experiment, we used the Limited Memory Variable Metric Algorithm implemented in the MegaM package.[6]

One of the drawbacks of this approach, however, is that it does not guarantee an object that is well-formed. Learning a probability distribution over EDUs and then choosing the most probable relation or attachment for each pair of EDUs potentially leads to structures that contain cycles. To avoid this, we can't blindly choose the most probable relation or attachment decision for each pair of EDUs. Instead, we should use this probability distribution as an input to a decoding mechanism.

6.1.4 Decoding with Maximum Spanning Trees

To answer our questions, "how many non-tree-like structures are there?" and "how far can tree decoding algorithms get us in multi-party dialogue?", our first decoder starts from the hypothesis that although the structures that we have are not trees, they can nonetheless roughly be approximated by them. To this end, we have started from the classic Maximum Spanning Tree (MST) algorithm— used by McDonald et al. (2005) for syntactic dependency parsing, as well as Muller et al. (2012b) and Li et al. (2014) for discourse parsing—tweaking it in order to produce structures that are closer to the ones specific to multi-party dialogue. The formal optimization problem is defined as follows:

$$T^* = \underset{T \text{ a spanning tree of } G}{\operatorname{argmax}} \sum_{e \in E(T)} w(e)$$

$$w(e) = \log\left(\frac{P_A(a|e)}{1 - P_A(a|e)}\right)$$

G being the complete graph of possible edges returned by the classifiers ; $E(D)$ representing the edges of D. The weight function w computes the log-odds of the attachment probability $P_A(a|e)$ returned by the MaxEnt attachment model. The edges of the spanning tree are then assigned the label with the highest probability $l^* = \operatorname{argmax}_l P_L(l|e)$ by the labeling model.

We used the Chu-Liu-Edmonds version of the MST algorithm (Chu and Liu, 1965; Edmonds, 1967), which 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 created an artificial node as the root with special dummy features. At the end of the procedure, this node points the real root of the discourse graph.

[6] Available from `http://www.cs.utah.edu/~hal/megam/`. We used the default regularization prior provided by MegaM.

Combining intra- and inter-turn models with the turn constraint As described above, we trained separate local models for intra- and inter-turn EDUs.

To create a full discourse tree, we perform a first decoding step, with the intra-turn model, to each turn in isolation. The graph from which we obtain the spanning tree is complete, as there are no restrictions on which EDUs can be linked inside a turn. We obtain the internal structure of each turn in the dialogue. It being a rooted tree, turns now possess a *discourse head*.

We then apply a second decoding step to detect edges between EDUs belonging to distinct turns. However, the graph from which we obtain a spanning tree only contains "forwards" edges (per the previously cited turn constraint), and only between the *discourse heads* of the turns. We obtain the full structure of the dialogue, with the sub-structures of the turns being joined together by their heads.

6.1.5 Experiments and results

Table 6.2 shows our results on our unseen test corpus, which contains a randomly selected 10% of dialogues in our corpus. The best configuration was selected after performing ten-fold cross validation on the training corpus. The reported results implement the turn-constraint during training for the local models. In other words, training instances for the local models include only forward links.

Along with MST decoding, we used the following two baselines:

- LAST, which always attaches an EDU to the preceding one, forming a single chain; despite its simplicity, this method is a very strong baseline in discourse parsing (Muller et al., 2012b, for example);

- LOCAL, a naive classifier that performs binary decisions on attachment and labeling based on the local probability model; attaching a pair of EDUs whenever $P_a(e) > 0.5$, then selecting the label with highest probability.

The LAST method gives us an F-score of 0.584 for attachment and 0.391 when we add the relations as well. The naive LOCAL method gives 0.541 for attachment and 0.446 for attachments and relations.

The best results for the global parsing problem exploited the turn constraint both during learning the local model and during decoding. Within a turn, our discourse structures are simple and largely linear; the best intra-turn results came from using LAST. Most of our interlocutors did not create elaborate discourse structures with long-distance attachments within the same turn. The inter-turn level was a different story, as the figures show. For inter-turn and the global problem, MST using the heads of the intra-turn substructures computed with LAST, produced the best results.

The obtain a 0.671 F1 score for unlabeled structures, and 0.516 for labeled structures.

To enable a comparison with RST style parsing where exact arguments for discourse relations are not computed, we achieved a F1 score of 0.68 for the task of undirected attachment in the full structure.

Despite the inherent difficulty of discourse parsing on multi-party chat dialogues (simultaneous, multiple discussion threads, improper syntax) our results are close to or better than the current state of the art for discourse parsing on monologue. We compare these results with two other approaches using dependency parsing strategies for discourse. Li et al. (2014) report an accuracy of 0.7506 for unlabeled structures and 0.4309 for the full labeled structures. Muller et al. (2012b) report 0.662 for unlabeled structure and 0.361 for labeled structures. We outperform both systems for fully labeled structures, and in spite of our non-tree-like structures we improve on them on unlabeled attachments. Though comparisons across different corpora are difficult, the numbers suggest that our results are competitive. Our results also suggest that one can get quite far with tree-based decoding algorithms, though we know that in principle MST cannot do better than 91% even with a oracle local model (ideal model in which an arc is giving probability 1 in case it occurs in the gold standard annotation).

Method	Undirected Attachment			Directed Attachment			Full Labeled Structure		
	prec	rec	F1	prec	rec	F1	prec	rec	F1
Last	0.602	0.566	0.584	0.602	0.566	0.584	0.403	0.379	0.391
Local	0.698	0.488	0.574	0.623	0.478	0.541	0.513	0.394	0.446
Intra-turn	0.837	0.955	0.892	0.808	0.922	0.861	0.489	0.558	0.521
Inter-turn	0.617	0.516	0.562	0.616	0.514	0.561	0.492	0.411	0.448
Global	0.697	0.663	0.680	0.688	0.655	0.671	0.529	0.503	**0.516**

Table 6.2: Evaluation results.

6.2 Directed acyclic graph decoding

We discussed in section 4.2 the particularities of multi-party dialogue. We know for a fact, that tree-based methods try to predict the *wrong* type of structure. For a dialogue containing n EDUs, a tree will include exactly $n-1$ discourse relations between them, even if the local model assigns high probability to additional relations. The *Settlers* corpus demonstrates that dialogues usually contain more than $n-1$ relations, so that a tree-based method, even with perfect precision, will not have perfect recall (as previously mentioned: 91% is the maximum attainable).

However, getting closer to the accurate structure of dialogue raises some issues, discussed in the next subsection, requiring the simplification of our original annotations.

6.2.1 From SDRT to dependency graphs

Motivation. 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 (i.e. Complex Discourse Units). We described in section 6.1.1 the *head replacement strategy* (HR) for eliminating CDUs from SDRSs.

However, 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 has 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 :

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

The correct SDRS for example 6.1 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 top graph in figure 6.2, which HR converts into the bottom one.

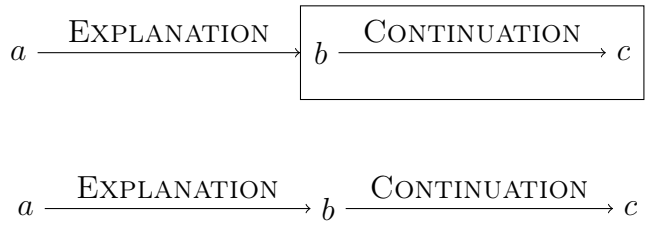


Figure 6.2: SDRS for example 6.1, and its HR conversion.

HR on example 6.1 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 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 example 6.1 slightly:

(6.2) [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 example 6.2, the SDRS graph would be the top graph in figure 6.3, which HR converts to the bottom one.

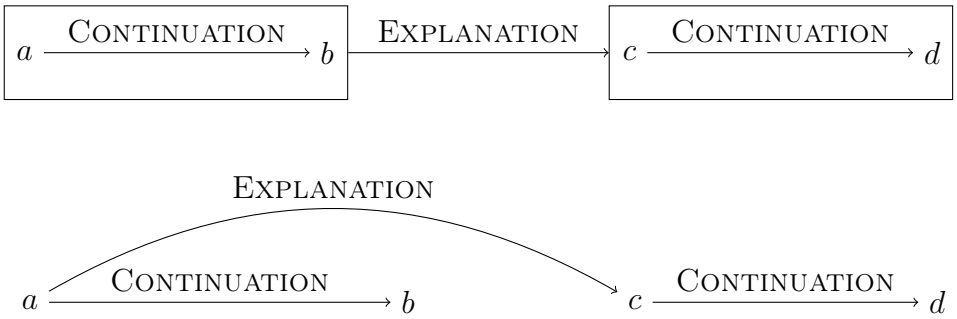


Figure 6.3: SDRS for example 6.2, and its HR conversion.

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 6.4 we show an example of how relations distribute between EDU/CDU, CDU/EDU and CDU/CDU.

The three strategies. This analysis of the conversion of SDRT graphs into dependency graphs leaves us with the following three CDU *replacement strategies*:

- HEAD: the original head-based strategy used in previous literature;

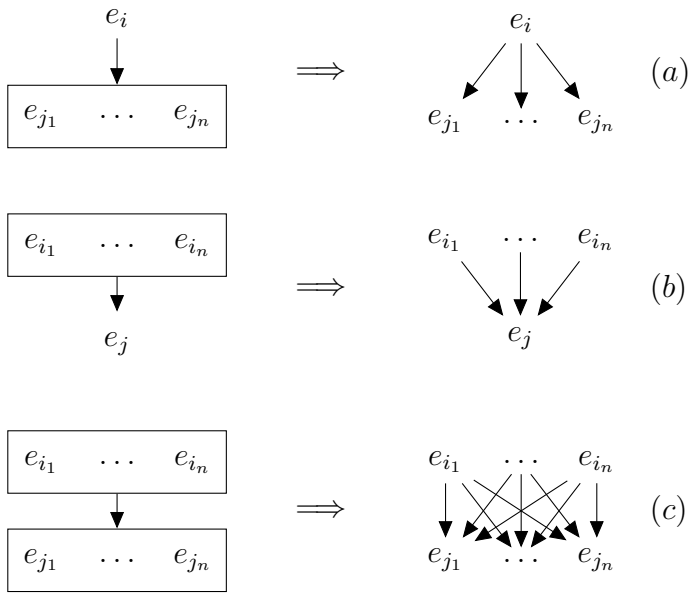


Figure 6.4: Distributing relations: (a) *right* distribution from an EDU to a CDU, (b) *left* distribution from a CDU to an EDU, (c) from a CDU to a CDU, assuming that the relations are distributive in their respective examples.

- PARTIAL: the distribution of edges according to the semantics provided by SDRT, discriminating between left, right, or either-distributive relations;
- FULL: the distribution of all edges, regardless of their label, considering all relations as left and right-distributive.

We conducted our experiments on three converted versions of our original SDRT corpus, one for each strategy.

6.2.2 Decoding with Integer Linear Programming

Formal definition An *integer linear programming problem* is a mathematical optimization problem, where the goal is to maximize (or minimize) a real-valued function over a defined set of variables, under a defined set of variables. More specifically, in ILP, the variables must be *integers* and the constraints must be *linear*, hence its name. The canonical form of an ILP problem is:

$$\begin{aligned} \text{maximize} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \\ & x \geq 0 \\ \text{and} \quad & x \in \mathbb{Z}^n \end{aligned}$$

where b, c are real-valued vectors and A is an integer-valued matrix.

A large number of problems can be formulated as ILP problems, such as the eight queens puzzle, the traveling salesman problem, many packing problems, etc. The resolution or general ILP problems is NP-hard, however entirely feasible when using a reasonably limited number of variables.

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.

Because we have left the domain of trees, well-explored by syntactic analysis and the previous works on discourse parsing, we must design new constraints on discourse graphs, which we have developed from empirical study of our corpus while also being guided by theoretical principles.

Objective function Our goal is to build the directed, edge-labeled graph $G = (D, A)$ stemming from the conversion of SDRT structures to dependency graphs. As a reminder:

- $D = \{e_1, \ldots, e_n\}$, the EDUs of the dialogue;
- $A \subset D \times D \times R$, the labeled edges, with R as the set of SDRT relation labels;
- if $(e_i, e_j, r) \in A$ then $\forall r' \neq r$, $(e_i, e_j, r') \notin A$ (i.e. only one label per edge).

Vertices in D are indexed from 1 to n, by their position in textual order. The labels are indexed from 1 to m in arbitrary order.

The local model provides us with two real-valued functions, which correspond closely[7] to the probabilities output by the MaxEnt local model. We note that taking the log-odds of the probabilities, as with MST decoding, did not have a significant effect on the results.

$$
\begin{gathered}
s_a : \{1, \ldots, n\}^2 \mapsto [0, 1] \\
s_a(i, j) \approx P_A(a|(e_i, e_j)) \\
s_r : \{1, \ldots, n\}^2 \times \{1, \ldots, m\} \mapsto [0, 1] \\
s_r(i, j, k) \approx P_L(k|(e_i, e_j))
\end{gathered}
$$

$s_a(i, j)$ 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

[7] Give or take a rounding error.

the n^2 binary variables a_{ij}, the mn^2 binary variables r_{ijk}, and the core constraint linking them:[8]

$$\begin{aligned} a_{ij} = 1 \quad &\equiv \quad (i,j) \in V \\ r_{ijk} = 1 \quad &\equiv \quad R(i,j) = k \\ \forall i,j \ \sum_{k=1}^{m} r_{ijk} &= a_{ij} \end{aligned}$$

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.

Constraints We describe now the set of constraints for our graphs. Properties that do not follow evidently from the formal expression of the constraint, such as connectedness and acyclicity, are proven in appendix A.

The first source of constraints is 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, which corresponds to the topic or initial move for the whole dialogue or text. 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.

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

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 theoretical observation that we call the *turn constraint*, discussed earlier in section 6.1.2. 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 the forward direction.

[8]Which is *a given edge, if it exists, has only one label.*

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 (cataphoric links).

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 labeling during training and testing.

Given the turn constraint we only need to ensure acyclicity of the intra-turn sub-graphs. 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:

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

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. 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.[9]

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.

[9] δ takes the values 1.0, 1.2 and 1.4 for the HEAD, PARTIAL and FULL distribution of the relations, respectively.

We compensate for this effect by using an additive cap $\eta \geq 0$ on the edge count, also estimated on the development corpus:[10]

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

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:[11]

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

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. Typically also, 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, more directed discourse moves and threads involving subgroups of the participants appear, but once again in negotiation dialogues it never happens that our participants return again and again to the same contribution. The state of the game evolves constantly, and older contributions quickly become irrelevant to the current situation. 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; EDUs within a sequence of contributions by the same speaker in our corpus are linked at least to the previous EDU (according to our previous experiments in section 6.1.5, this is a reasonable hypothesis); finally, that edges with zero score are not included in the graph:

[10] η takes the value of 4 for the FULL distribution while it has no upper bound for the HEAD and PARTIAL distributions.

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

$$
\begin{aligned}
\forall i \quad & a_{ii} = 0 \\
\forall i \quad & t(i) = t(i+1) \implies a_{i,i+1} = 1 \\
\forall i, j \quad & s_a(i,j) = 0 \implies a_{ij} = 0 \\
\forall i, j, k \quad & s_r(i,j,k) = 0 \implies x_{ijk} = 0
\end{aligned}
$$

6.2.3 Experiments and results

As with MST decoding (cf. section 6.1.3), 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 their dialogue acts (whether they involved an offer, a counteroffer, etc).

The MaxEnt model itself was trained, this time, using the *scikit-learn* library (Pedregosa et al., 2011). For ILP decoding, we used the SCIP optimization suite (Gamrath et al., 2016).

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 and only if the probability of attachment between EDUs i and j is higher than 0.5.

Each of the three distribution methods described at the end of section 6.2.1 (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 6.3 shows the main statistics of our dataset. The influence of the CDU replacement strategies is clearly visible: the more relations are distributed, the higher the relation count in the converted dataset.

Table 6.4 shows our evaluation results, comparing decoders and baselines for each of the distribution strategies.

	Total	Training	Testing
Dialogues	1091	968	123
Turns	9160	8166	994
EDUs	10677	9545	1132
CDUs	1284	1132	152
Relation instances			
HEAD	10191	9127	1064
PARTIAL	11734	10507	1227
FULL	13675	12210	1465

Table 6.3: Dataset overview

As can be seen, our ILP decoder consistently performs significantly better than the baselines as well as our own MST decoder, even when restricted to tree structures and HEAD strategy (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 labeling 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 the margin between ILP's performance and other methods considerably widened by increasing the edge count of the target structures. 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.

Decoder	Model	Unlabeled Attachment			Labeled 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 6.4: Evaluation results.

Chapter 7

Parsing argumentative structure

Many works in the field of argumentative parsing mention the similarities between discourse and argumentation.[1] However, while parsing methods from the two domains share common techniques, an in-depth comparison of the structures of discourse and argumentation needed annotated material enabling this research.[2] In Stede et al. (2016), which section 7.1 draws from, we set out to create a dataset suitable for the task.

After having studied methods to extract discourse structure, our next step was to transfer our Integer Linear Programming methods to argumentation parsing, expanding of the work of Peldszus and Stede (2015). Our approach, described in section 7.2, is currently unpublished.

7.1 Building a parallel corpus

We described, in chapter 2, three approaches to analyzing and representing discourse structure have resulted in various annotated corpora and in implemented discourse parsers:

- The Penn Discourse Treebank (PDTB) annotates individual connectives with their coherence relations and their argument spans (Prasad et al., 2008).

- Rhetorical Structure Theory (RST) predicts tree structures on the grounds of underlying coherence relations that are mostly defined in terms of speaker intentions (Mann and Thompson, 1988).

[1] In contrast, we barely found any mention of argumentation frameworks in the discourse parsing literature.

[2] In argumentation as well, data is scarce.

- Segmented Discourse Representation Theory (SDRT) exploits graphs to model discourse structures and defines coherence relations via their semantic effects on commitments rather than relative to speaker intentions (Asher and Lascarides, 2003; Lascarides and Asher, 2009).

As we plan to study, in this chapter, the full argumentative structure of text, our concern is with RST and SDRT only. To date, it has been difficult to compare the two frameworks on empirical grounds, since there were no directly-comparable parallel annotations of the same texts. To improve upon this situation, we took an existing corpus of 112 short "microtexts", which had already been annotated with argumentation structure, and added layers for RST and SDRT. To this end, we harmonized the underlying segmentation rules for minimal discourse units, so that the resulting structures can be compared straightforwardly. We implemented an approach to merge the annotations, and we report here some initial observations on the correlations between RST, SDRT and argumentation in that corpus.

In addition to comparing RST and SDRT, we foresee interesting applications of this kind of corpus data for purposes of argumentation mining. The correlations between discourse structure and argumentation structure have not been studied yet in depth, and thus it is not clear whether established discourse parsing techniques (geared either toward RST or toward SDRT) can contribute to an automatic argumentation analysis, and if so, in what ways.

In the following, we introduce our data set (section 7.1.1) and describe the three layers of annotation (section 7.1.3). Then, we explain the mapping of the layers to a common dependency tree format (section 7.1.4), and we present some initial observations on correlations (section 7.1.5).

7.1.1 Argumentative texts

The "corpus of argumentative microtexts" (Peldszus and Stede, 2016), henceforth referred as the *Microtext* corpus, has been designed as a collection of relatively "simple" yet authentic texts enabling the study of argumentation. It consists of 90 texts that have been collected in a controlled text generation experiment, where 23 competent subjects wrote short texts of controlled linguistic and rhetoric complexity, discussing one of the issues they chose from a pre-defined list of controversial issues. These include questions like "Should everybody be required to pay fees for public radio and TV" or "Should health insurers cover alternative medical treatments".

Each text was to fulfill three requirements: it should be about five segments long; all segments should be argumentatively relevant, either formulating the main claim of the text, supporting the main claim or another segment, or attacking the

Should health insurers pay for alternative treatments?
Health insurance companies should naturally cover alternative medical treatments. Not all practices and approaches that are lumped together under this term may have been proven in clinical trials, yet it's precisely their positive effect when accompanying conventional 'western' medical therapies that's been demonstrated as beneficial. Besides many general practitioners offer such counselling and treatments in parallel anyway - and who would want to question their broad expertise?

Figure 7.1: Sample text from the *Microtext* corpus.

main claim or another segment. Also, the writers were asked that at least one possible objection to the claim should be considered in the text.

To supplement the original German version of the collected texts, the whole corpus has been professionally translated into English. Figure 7.1 shows a sample text from this English part of the corpus. A more detailed overview of the data collection is given in Peldszus and Stede (2016).

For the purposes of this study, we worked with the English version of the corpus. The finer EDU segmentation as well as the creation of the additional RST and SDRT annotation layers was done on the basis of the English text. Mapping the new annotations back to the German version of corpus will be achieved by future efforts. The corpus is freely available online.[3]

7.1.2 Aligned segmentation

In order to achieve comparable annotations on the three layers, we decided in the beginning of the project to aim at a common underlying discourse segmentation. For a start, the argumentation layer already featured ADU (*argumentative discourse unit*) segmentation; these units are relatively coarse, so it was clear that any ADU boundary would also be an EDU (*elementary discourse unit*) boundary in RST and SDRT. On the other hand, the discourse theories often use smaller segments. Our approach was to harmonize EDU segmentation in RST and SDRT, and then to introduce additional boundaries on the argumentation layer where required, using an "argumentatively empty" JOIN relation.

As explained in the next two sections, RST and SDRT annotation start from slightly different assumptions regarding minimal units. After building the first versions of the structures, we discussed all cases of conflicting segmentations and tweaked both annotations so that eventually all EDUs were identical.

[3]For the original German/English corpus, see `https://github.com/peldszus/arg-microtexts`. The finer segmented, multi-layer annotation done in this study for English is available at `https://github.com/peldszus/arg-microtexts-multilayer`.

The critical cases fell into three groups:

- "Rhetorical" prepositional phrases: Prepositions such as 'due to' or 'despite' can introduce segments that are rhetorically (and sometimes argumentatively) relevant, when for instance a justification is formulated as a nominalized eventuality. We decided to overwrite the syntactic segmentation criteria with a pragmatic one and split such PPs off their host clause in cases where they have an argumentative impact.

- VP conjunction: These notoriously difficult cases have to be judged for expressing either two separate eventualities or a single one. We worked with the criterion that conjoined VPs are split in separate EDUs if only the subject NP is elided in the second VP.

- Embedded EDUs: For technical reasons, the Potsdam Commentary Corpus Stede and Neumann (2014, in German) annotation had not marked center-embedded discourse segments; and, in general, different RST projects treat them in different ways. In SDRT, however, they are routinely marked as separate EDUs. In the interest of compatibility with other projects, we decided to build two versions of RST trees for texts with embedded EDUs: one version ignores them, while the other splits them off and uses an artificial "Same-Unit" relation to repair the structure (cf. Carlson et al. (2003) and section 2.1.1 of this work).

As a result of the finer segmentation, 83 ADUs not directly corresponding with an EDU have been split up, so that the final corpus contains 680 EDUs.

7.1.3 Structure annotation

Argumentation The initial release of the corpus already incorporated argumentation structures for all texts, following the scheme devised in Peldszus and Stede (2013), which itself is based on Freeman's theory of the macro-structure of argumentation (Freeman, 1991; Freeman, 2011). Its central idea is to model argumentation as a hypothetical dialectical exchange between the *proponent*, who presents and defends his claims, and the *opponent*, who critically questions ("attacks") them in a regimented fashion. Every move in such an exchange corresponds to a structural element in the argumentation graph (cf. section 3.2).

The first step in an analysis consists in segmenting the text into its argumentative discourse units (ADUs); these may in turn consist of several elementary discourse units (EDUs) as used in RST and SDRT. The argumentation structure scheme then distinguishes between simple support (one ADU provides a justification of another) and linked support, where several ADUs collectively fulfill the

role of justification. On the side of attacks, we separate rebutting (denying the validity of a statement) and undercutting (denying the relevance of a statement in supporting another). The scheme is designed in such a way that the fine-grained representations can be reduced to coarser ones that, for example, only distinguish between *support* and *attack* (see Peldszus and Stede (2015)), as it is customary in much of the related work on argumentation mining.

In Figure 7.2, we show the representation for the sample text given in Figure 7.1. The nodes of this graph represent the propositions expressed in text segments (grey boxes), and their shape indicates the role in the dialectical exchange: round nodes are proponent's nodes, square ones are opponent's nodes. The arcs connecting the nodes represent different supporting (arrow-head links) and attacking moves (circle/square-head links). By means of recursive application of relations, representations of relatively complex texts can be created, identifying the central claim of a text, supporting premises, possible objections and their counter-objections.

These structures have been annotated on the German texts by two experts, and they apply equally to the English translation. The guidelines are specified in Stede (2016). They have been shown to yield reliable agreement, see Peldszus (2014).

The annotated corpus contains 576 ADUs, of which 451 are proponent and 125 opponent ones. The most frequent relation is SUPPORT (263), followed by REBUT (108), UNDERCUT (63). LINKED relations (21) and support by EXAMPLE (9) occur only rarely.

RST The RST annotations have been created according to the guidelines (Stede, 2016) that were developed for the Potsdam Commentary Corpus (Stede and Neumann, 2014, in German). The relation set is quite close to the original proposal of Mann and Thompson (1988) and that of the RST website[4], but some relation definitions have been slightly modified to make the guidelines more amenable to argumentative text, as it is found in newspaper commentaries or in the short texts of the corpus we introduce here. Furthermore, the guidelines present the relation set in four different groups: primarily-semantic, primarily-pragmatic, textual, multinuclear. The assignment to 'semantic' and 'pragmatic' relations largely agrees with the subject-matter/presentational division made by Mann & Thompson and the RST website, but in some cases we made diverging decisions, again as a step to improve applicability to argumentative text; for example, we see EVALUATION as a pragmatic relation and not a semantic one. 'Textual' relations cover phenomena of text structuring; this group is motivated by the relation division proposed by Martin (1992), but the relations themselves are a subset of those of Mann &

[4]www.sfu.ca/rst

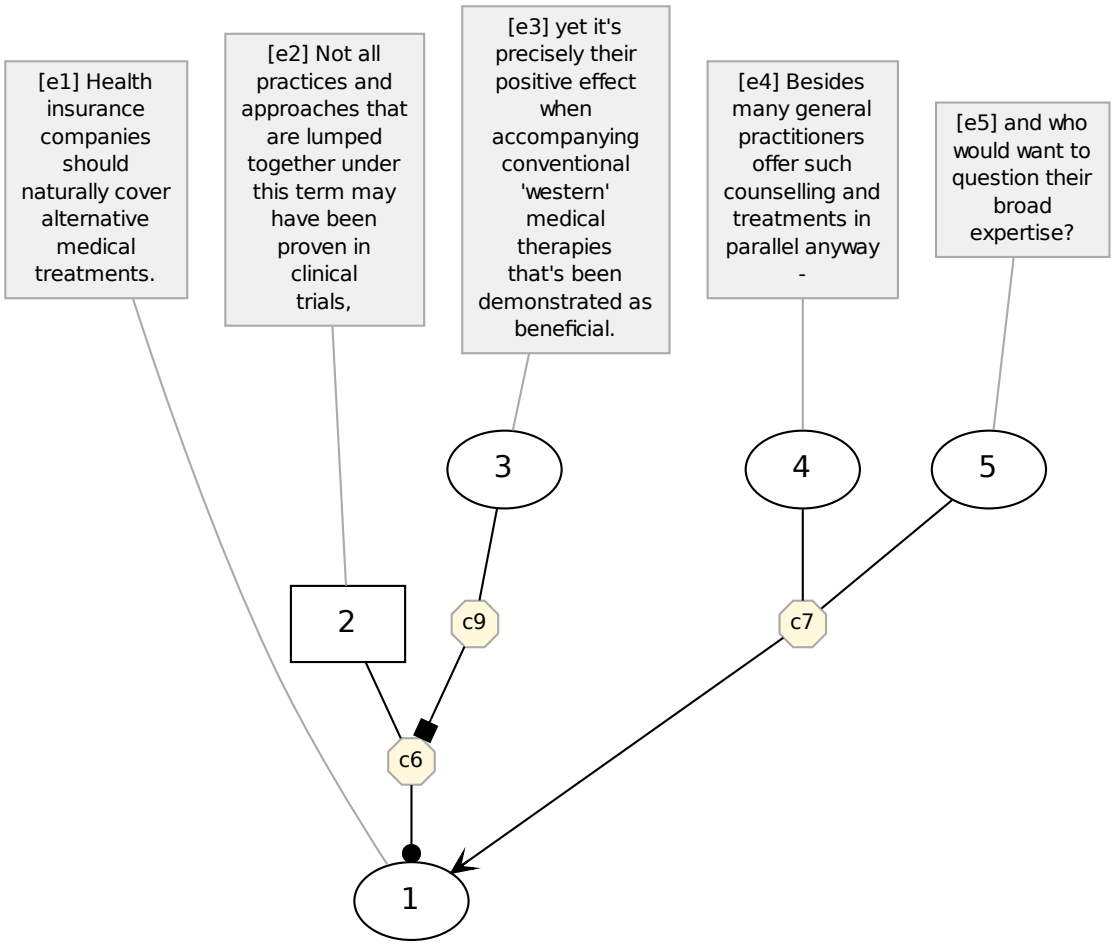


Figure 7.2: Argumentation structure of the example text. Here 4 and 5 support 1, 2 attacks 1, and 3 undercuts this attack.

Thompson and the website (e.g., List, Preparation). Finally, the 'multinuclear' relations are taken from the original work, with only minor modifications to some definitions.

The annotation procedure explained in the guidelines suggests to prefer pragmatic relations over semantic ones in cases of ambiguity or doubt, which is also intended as a genre-specific measure. All RST annotations on the Microtext corpus were done using the RSTTool[5]. In the resulting corpus, there are 467 instances of RST relations, hence on average 4.13 per text. The most frequent relation is (by a large margin) Reason (178 instances), followed by Concession (64), List (63), Conjunction (44), Antithesis (32), Elaboration (27), and Cause/Result (22); other relations occur less than 20 times.

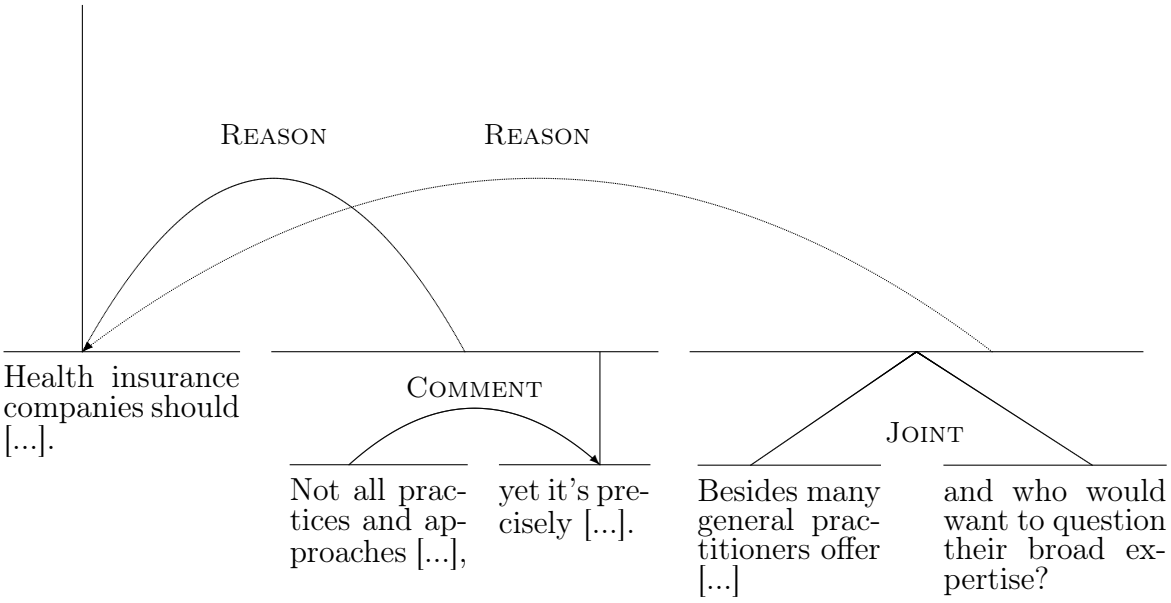


Figure 7.3: RST representation, with arrows pointing to the Nucleus.

SDRT The SDRT annotations were created following the ANNODIS annotation manual (Muller et al., 2012a) which was based upon Asher and Lascarides (2003). The amount of information about discourse structure was intentionally restricted in this manual. Instead it focused essentially on two aspects of the discourse annotation process: segmentation and typology of relations. Concerning the first, annotators are provided with an intuitive introduction to discourse segments, including the fact that we allowed discourse segments to be embedded in one another as well as detailed instructions concerning simple phrases, conditional

[5]http://www.wagsoft.com/RSTTool/

and correlative clauses, temporal, concessive or causal subordinate phrases, relative subordinate phrases, clefts, appositions, adverbials, coordinations, etc. Concerning discourse relations, the goal of the manual was to develop an intuition about the meaning of each relation. Occasional examples were provided, but we avoided an exhaustive listing of possible discourse markers that could trigger a particular relation, because many connectives are ambiguous and because the presence of a particular discourse connective is only one clue as to what the discourse relation linking two segments might be.[6] For the purposes of this annotation campaign we used the Glozz annotation tool.[7]

The SDRT corpus contains 669 EDUs, 183 CDUs and 556 relations. The most frequent relations are CONTRAST (144), ELABORATION (106), CONTINUATION (80), RESULT (76), EXPLANATION (55), PARALLEL (26), CONDITIONAL (23) while the rest had fewer than 20 instances. Figure 7.4 shows the SDRT graph for the text shown in Figure 7.1.

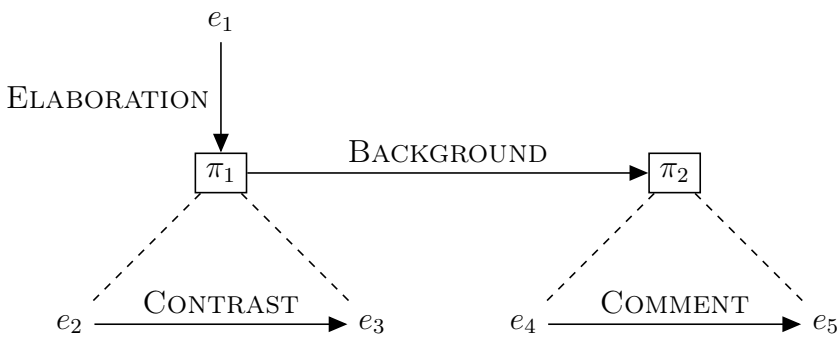


Figure 7.4: SDRT structure of the example text.

7.1.4 A common format: dependency structures

Calculating correlations between argumentation and discourse as well as between the two discourse corpora themselves requires converting the annotations from their tool-specific XML formats (RSTTool, Glozz) into a common format. This is not an easy task since the two theories have fundamental differences at least as far as scoping of relations is concerned. We consider dependency structures

[6]The manual also did not provide any details concerning the structural postulates of the underlying theory, including constraints on attachment (the so-called "right frontier" of discourse structure), crossed dependencies and more theoretical postulates. The goal of omitting such structural guidelines was the examination of whether annotators respected the right-frontier constraint or not (Afantenos and Asher, 2010).

[7]http://www.glozz.org

as a reasonable candidate for a common format capturing the structures of RST and SDRT, as it had also been proposed earlier by Danlos (2005). This is further facilitated by the fact that—with the exception of embedded EDUs in SDRT, for which we used the Same-Unit "relation" in RST—both annotations use the same EDUs.

In our case, dependency structures are graphs whose nodes represent the EDUs and whose arcs represent the discourse relations between the EDUs. Given this representation, calculating correlations between argumentation and discourse becomes an easy task since we have the same nodes, and only the relations vary.

Furthermore, future experiments on discourse parsing and argumentation structure analysis can be facilitated by using a common format for all annotations; however, we need to be cautious when it comes to theory-specific discourse parsing, since the mapping between the theories is not one to one.

SDRT makes use of CDUs to represent larger units of discourse. The problem of converting them to dependency graphs has been discussed previously in sections 6.1.1 and 6.2.1.

RST, on the other hand, makes use of some version of the "Nuclearity Principle" to determine what is the exact scope of a discourse relation. Most formulations of the Nuclearity Principle hinder a structural match between RST trees and SDRT graphs, as detailed in Venant et al. (2013). In this paper, the authors axiomatize that both RST trees and SDRT graphs in an ecumenical fragment of monadic second order logic, so that precise translation results can be proved concerning the posited structures of the two theories. They show that if one restricts SDRT graphs to those that have just one incoming arc to each node, then one SDRT graph may correspond to several RST trees. Nevertheless for the restricted and simplified texts of the argumentation corpus, it seems that the two structures are largely inter-translatable, depending on (i) how we translate CDUs into a dependency graph and (ii) how we fix the arguments of relations in the translation of an RST tree into a dependency graph.

Another obvious mismatch concerns the labels of the relations in the two theories. Because RST and SDRT start from different explanatory goals, they employ different principles for individuating their sets of discourse relations. For example, our analyses of the sample text in figures 7.3 and 7.4 show that an SDRT ELABORATION corresponds to REASON in the RST tree. Such differences can in principle be due to the different motivations of the theory (identify relations primarily on the basis of semantic properties of the argument, or on the grounds of interpreted speaker intentions), or they can result simply from different readings of the text by the respective analysts. Clarifying this in our corpus, and undertaking more principled comparisons between the theories is one goal for our future work with the aligned corpora.

7.1.5 Comparison between annotation layers

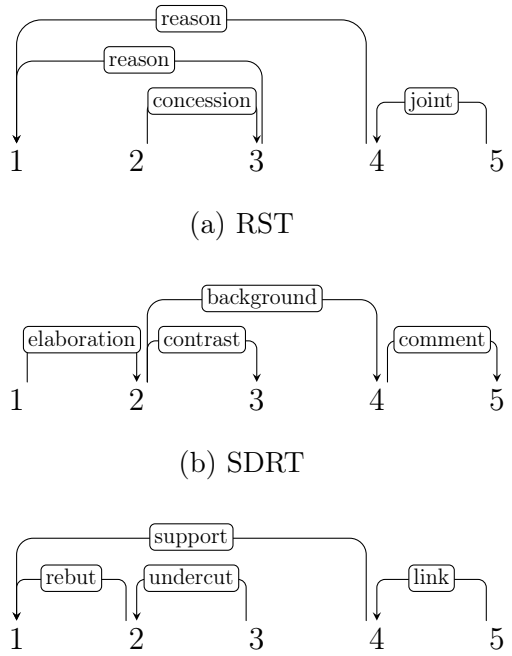


Figure 7.5: Example dependency conversions for the example text from the annotations of the three theories.

Methodology The parallel annotation of the corpus converted to a dependency format now invites systematic comparison of the three structures. As we can see in figure 7.5, there are evident structural similarities between discourse structures— both RST and SDRT— and argumentative structure. Segment 1 holds the most prominent position in the SDRT graph, is the central nucleus in the RST tree, and the "main thesis" in the argumentation. The proponent/opponent distinction made in the argumentation analysis (circle vs. box node) of course has no direct counterpart in RST and SDRT, but the perspective switch between the two roles might be indicated by adversarial coherence relations. For a quantitative, pairwise comparison of the correspondences between related segments and the relation types, we apply two strategies: *common edges*, and *common connected components*.

First, we look for undirected edges common to the different structures. In the example shown in figure 7.5, an edge between 2 and 3 and between 4 and 5 is found in all structures. Note that the first ones all have an adversative relation label, while the latter all have a more organizational relation label assigned. Argumentation and SDRT share an edge between 1 and 2, while argumentation and

RST share an edge between 1 and 4. For the purpose of quantitative comparison, we collect the relations of all common edges in a co-occurrence matrix. Edges in one graph without a correspondence in the other graph are mapped to *none* in this matrix. An example matrix for argumentation and RST is shown in table 7.1 and will be discussed below.

Furthermore, we extend the scope of analysis and look for connected components common to both structures. We apply a simple sub-graph alignment algorithm yielding connected components with 2, 3 or 4 nodes occurring in the undirected, unlabeled graphs of both structures. This can reveal typical structural patterns. We can then determine how often these matches can be successfully mapped to one another given the relation labels. The structures shown in figure 7.5 have for example several common components: All of them share a sub-graph 1, 2, 3, although with different connection configurations. RST and argumentation additionally share a sub-graph 1, 4, 5, with aligned connections. We will sum over the corpus, how often these common sub-graphs occur and how likely they can be mapped to each other based on the relations.[8]

Argumentation vs. RST The co-occurrences of the edge-labels are shown in table 7.1. In total, 60% of the edges are common in both structures. The most frequent class of SUPPORT edges in argumentation correspond mainly with REASON and some CAUSE and EVIDENCE edges, however 39% of them do not map to RST edges. The second frequent class in argumentation, REBUT, does not map well to RST: 72% of those edges have no correspondence in RST. The rest co-occurs with ANTITHESIS and CONCESSION. A very wide distribution of RST relation labels is found for the JOIN relation in argumentation. This relation connects multiple EDUs to argumentatively relevant ADUs and is converted to dependencies in a left-to-right fashion. Since the nucleus in RST is not necessarily the left-most node, it correlates with both less argumentative relations such as CONJUNCTION or CONDITION and more argumentative relations such as REASON or CAUSE. For the argumentative UNDERCUTs, most of them align with CONCESSION and ANTITHESIS, while 33% do not co-occur with RST relations. Note, that nearly no correspondence can be found for RST LIST relations.

Regarding the common components in both theories, about 43% of all 3 node argumentation sub-graphs can be matched to RST sub-graphs, and 46% vice versa. Most of them are parallel structures, e.g. 2 SUPPORTs for a claim on the argumentation side and two parallel REASONs on the RST side. On the other hand there are also common sub-graphs with differing edges, e.g. when the argumentation structure features two separate SUPPORTs or REBUTs, while the RST structure

[8]Note that the comparisons in this subsection exclude 8 texts with center-embedding, as these complicate the correlation procedure here.

joins them into one larger span in a LIST or CONJUNCTION. Very interesting are the attack- and counter-attack constructions, some of which are shown in figure 7.6. The RST annotations do not explicitly represent the rebutting functions of segments, but instead take the counter-attack as a reason for the claim. While the countering of an attack is implicitly supporting the attacked claim, supporting a claim cannot be taken as an implicit counter of potential attacks. The RST structure is thus missing one aspect of the attack- counter-attack structure.[9] This also become evident by the different predictive power of this correspondence. For the linearization with the claim first, the argumentation structure 7.6c can be mapped to the RST structure 7.6d in 81%, but vice versa only in 60%. For the linearization with the claim behind, the situation is less clear: The argumentation structure 7.6a can be mapped to the RST structure 7.6b in 57%, vice versa in 67%. A more detailed comparison of the different sub-graph correspondences is left for future work.

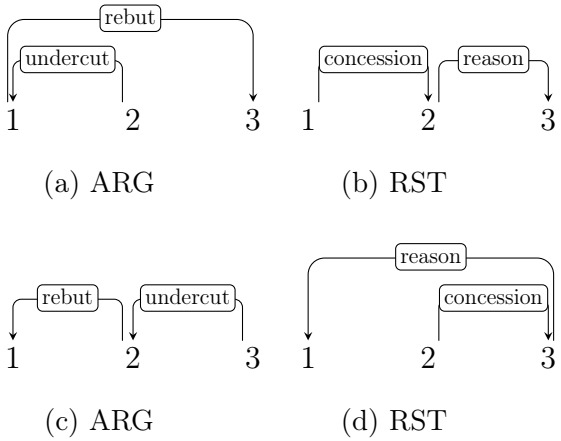


Figure 7.6: Common components between RST and ARG for attack-, counter-attack constructions.

Argumentation vs. SDRT When comparing common edges, we find that 63% of the edges can be mapped from one structure to the other. The co-occurrences of the relation labels are shown in figure 7.2. Argumentative SUPPORTs co-occur with ELABORATION, EXPLANATION, and RESULT. However, 48% of the supports cannot be mapped to SDRT edges, which is more than in the alignment of argumentation and RST. REBUTs correspond mainly with CONTRAST, but also with ELABORATION, the remaining 43% of the rebutting edges do not map to SDRT,

[9]This point was already raised by Peldszus and Stede (2013), but could only now be investigated on a larger empirical basis.

	example	join	link	rebut	support	undercut	NONE
antithesis	.	3	.	9	1	6	7
background	.	1	2	.	4	.	8
cause	.	4	1	.	11	.	2
circumstance	.	4	.	.	.	.	1
concession	.	.	.	6	1	32	18
condition	.	13	.	1	1	.	.
conjunction	.	10	6	.	.	2	23
contrast	.	.	.	1	.	.	3
disjunction	.	2	.	.	.	.	2
e-elaboration	2	5	.	.	.	.	1
elaboration	4	7	.	2	3	.	11
evaluation-s	.	2	.	.	.	.	.
evidence	.	.	.	.	8	.	2
interpretation	.	.	.	.	.	.	2
joint	.	2	5	1	4	1	8
justify	.	.	.	.	4	.	3
list	.	1	.	1	2	.	53
means	.	1	.	.	.	.	.
motivation	.	.	.	1	2	.	.
preparation	.	3	.	.	.	.	.
purpose	.	3	.	.	.	.	.
reason	.	6	.	3	99	.	55
restatement	.	.	.	.	2	.	2
result	.	1	.	.	1	.	.
sameunit	.	1	.	1	.	.	.
solutionhood	.	.	.	.	.	.	1
unless	.	.	.	1	.	.	1
NONE	2	10	7	72	92	20	.

Table 7.1: Co-occurrence matrix for edge labels for RST (rows) vs Argumentation (columns).

which is better than the coverage of RST for this relation. Undercutting attacks are quite clearly related to CONTRAST. As in RST, instances of the JOIN relations in argumentation structures distribute widely over the SDRT relations. From the SDRT perspective it is striking that nearly no correspondence is found for CONTINUATION relations. Also, 34% of the CONTRAST relations do not align with edges in the argumentation graphs.

Looking at the common components, we cannot only investigate larger sub-graphs but also consider the direction of the edges. Forward-looking supports (i.e. 1 supports 2) rather map to RESULT, while backward-looking supports (i.e. 2 supports 1) rather correspond with ELABORATIONs. EXPLANATIONs can be found for both directions of supports. In a similar vein, ELABORATIONS co-occur with REBUTs only, when the latter are backward-looking, not when the rebutted claim comes after the rebuttal. CONTRASTs can be found for both directions of rebuttal.

For larger sub-graphs with 3 nodes, 49% of the argumentation graphs can be mapped to SDRT, vice versa 53%. The most frequent correlation shown in figures 7.7a and 7.7b. The common REBUT & UNDERCUT scheme in argumentation only maps to SDRT when linearized backward-looking. The SDRT correspondence of two CONTRASTs, as shown in figures 7.7c and 7.7d, is only found in 35%, the remaining instances leave either the adversative character of the rebuttal or of the undercutter underspecified by using other relations such as ELABORATION, EXPLANATION or CONDITIONAL. As in RST, the identification of argumentative attacks and counter-attacks by chains of adversative relations is not trivially achieved and might require a deeper investigation of the surrounding signals.

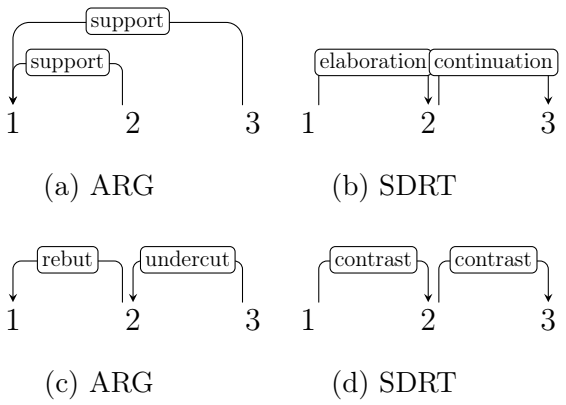


Figure 7.7: Common components between ARG and SDRT

	example	join	link	rebut	support	undercut	NONE
alternation	.	1	.	1	.	1	4
background	.	3	.	.	4	.	1
comment	1	2	2	.	2	.	2
conditional	.	12	.	3	.	1	1
continuation	1	2	1	.	5	1	62
contrast	.	6	1	35	6	39	45
e-elab	1	3	.	.	.	.	.
elaboration	4	8	3	10	46	.	26
explanation	.	4	1	2	33	.	6
frame	.	4	1	.	1	.	.
goal	.	.	1	.	.	.	.
narration	.	3	1	.	.	1	2
parallel	.	5	2	.	1	4	13
result	.	16	2	5	25	.	23
NONE	1	10	6	43	112	14	.

Table 7.2: Co-occurrence matrix for edge labels for SDRT (rows) vs Argumentation (columns)

7.2 Parsing the *Microtext* corpus

7.2.1 Local models

In order to perform structured output prediction on argumentation structures, ideally what one would like to do is to learn a model

$$h : \mathcal{X}_{A^n} \mapsto \mathcal{Y}_{\mathcal{G}}$$

where $\mathcal{X}_{A^n}$ is the domain of instances representing a collection of ADUs for each dialogue and $\mathcal{Y}_{\mathcal{G}}$ is the set of all possible argumentation graphs. Directly predicting argumentation structures, though, is a very difficult task which requires an amount of data that we currently lack in the community since, in a sense, every document is considered as a single instance. Moreover no appropriate logistic or hinge loss function (Smith, 2011) has been proposed in the community either for argumentation or discourse structures. Most approaches, including our novel ILP approach, aim thus at the more modest goal of learning a model

$$h : \mathcal{X}_{A^2} \mapsto \mathcal{Y}_R$$

where the domain of instances $\mathcal{X}_{A^2}$ represents features for a pair of ADUs and $\mathcal{Y}_R$ represents the set of argumentative relations. The upshot of this is that we are

building a local sort of model that yields a probability distribution of relations between individual ADUs.

Note that we do not directly make a classifier out of this model. In other words, we do not try to directly extract relations from the above model by searching for a threshold that will have optimal local results. Indeed, concatenating the relations predicted by a local classifier would not necessarily yield a well-formed structure, even with good *local* results; there would be no guarantee that there would be no cycles or a single connected component, as required by our data. Instead we use the probability distribution that this model yields as input to a decoder that tries to optimize a *global* measure of the argumentation structure.

Dependency Structures We use the dependency conversion of the argumentative portion of the corpus, as presented in section 7.1.4, with the coarse grained set of relations $\{support, attack\}$.

For illustration, figure 7.8 shows the dependency graph for the argumentation structure of the following example.

(7.1) [Health insurance companies should naturally cover alternative medical treatments.]$_1$[Not all practices and approaches that are lumped together under this term may have been proven in clinical trials,]$_2$[yet it's precisely their positive effect when accompanying conventional 'western' medical therapies that's been demonstrated as beneficial.]$_3$[Besides many general practitioners offer such counselling and treatments in parallel anyway -]$_4$[and who would want to question their broad expertise?]$_5$

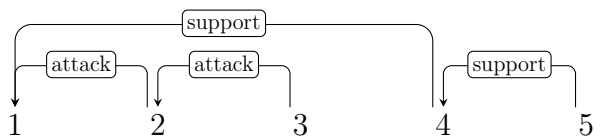


Figure 7.8: Dependency conversion of the argumentation structure of example 7.1

Subtasks Peldszus and Stede (2015) proposed the following four subtasks for predicting the argumentation structures:

- **attachment (at)**: Given a pair of ADUs, are they connected by an argumentative relation? [yes, no]
- **central claim (cc)**: Given an ADU, is it the central claim of the text? [yes, no]

- **role (ro)**: Given an ADU, is it in the [proponent]'s or the [opponent]'s voice?
- **function (fu)**: Given an ADU, what is its argumentative function? [support, attack, none]

We reproduced this approach and trained a log-loss SGD (stochastic gradient descent) classifiers for each of these tasks. Note, that relation labels are classified using only the source segment. We reimplemented their feature set, which includes lemma uni- and bigrams, the first three lemmas of each segment, POS-tags, lemma- and POS-tag-based dependency parse triples, discourse connectives, main verb of the sentence, and all verbs in the segments, absolute and relative segment position, length and punctuation counts, linear order and distance between segment pairs.

For the syntactic analysis, we use the spaCy parser (Honnibal and Johnson, 2015) instead of the mate parser (Bohnet, 2010). Both parsers provide pretrained models for English and German. The spaCy parser is a bit less accurate and does not offer a morphological tagging, but it is very fast and allows us to greatly simplify the pipeline. Moreover, it comes with Brown clusters and vector-space representations, which we want to test. Another difference is that we extended the lexicon of English discourse connectives with the connectives collected in the EDUCE project.[10]

New features In addition to the reimplemented feature set, we test the impact of the following new features: We add Brown cluster unigrams (BC) and bigrams (BC2) of words occurring in the segment. We completed the discourse relations features (DR): While the lexicon of discourse connectives for German used in experiments of Peldszus and Stede (2015) was annotated with potentially signaled discourse relations, their English lexicon was lacking this information. We extended the English connective lexicon by those collected in the EDUCE project which also have been annotated with signaled discourse relations. Also, a feature representing the main verb of the segment was added; the already existing verb features either focused on the verb of the whole sentence which might be too restrictive, or on all possible verbal forms in the segment which might not be restrictive enough.

In order to investigate the impact of word embeddings for this task, we add the 300 dimensional word-vector representations, averaged over all content words of the segment, as a feature for segment wise classifiers (VEC). Stab and Gurevych (2016) gained small improvements –around 1 point F1-score on their dataset– by adding word-embeddings as a features to their argumentative stance classifier. Moreover, we derive scores of semantic distance between two segments using these vectors: We measure the cosine distance between the average word vector representations

[10] `https://github.com/irit-melodi/educe`

of the segment and its left and right antecedents (VLR). Also, for the attachment classifier, we measure the cosine distance between the average word vectors of the source and target segment (VST).

Furthermore, we added features for better capturing the inter-sentential structure, i.e. for relations with subordinate clauses: One feature representing that the source and target segments are part of the same sentence (SS) and one representing that the target is the matrix clause of the source (MC).

7.2.2 Decoders

MST decoder In a classic MST decoding scenario, one uses a matrix $\mathbf{\Pi} \in \mathbb{R}^{n \times n}$ representing the attachment probability distribution of the local model. The Chu-Liu-Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967) is then used in order to find the maximum spanning tree. The predicted edges could finally be labeled in a subsequent step using a separate classifier.

In contrast to that, Peldszus and Stede (2015) opt in jointly predicting attachment and the other levels using MST methods. They first set up a fully connected multigraph with as many parallel edges as relations-types (in their case two, for supporting and attacking relations). This is the "evidence graph" in their terminology. A local model is trained for each of the four levels (attachment, central claim, role and function). From the scores of the local models, four probabilities are derived which are linearly combined into one edge score in the multigraph: the probability of attachment, the probability of having the corresponding argumentative function, the probability of the source not to be the central claim and the probability of switching the argumentative role from the source to the target segment (for attacks) or of preserving it (for supports). The multigraph is reduced to a graph, for which the maximum spanning tree is found. The combination of these probabilities constrains some typical interactions between the different levels in the argumentation structure.

We replicate this decoder using the exact same procedure and the results of the local models described in section 7.2.1.

Novel ILP decoder Using as input the same local model as used before, we try to build a directed acyclic graph $G =< V, E, R >$. Vertices (ADUs) are referred by their position in textual order, indexed starting from 1. The argumentative functions $central\_claim, attack, support$ are referred by their respective indexes $\nu_{cc} = 1, \nu_a = 2, \nu_s = 3$. Let $n = |V|$. We create four sets of core variables

corresponding to the levels of prediction:

$$
\begin{aligned}
cc_i &= 1 \equiv adu_i \text{ is a central claim} \\
ro_i &= \begin{cases} 1 & \text{if } adu_i \text{ is a proponent node} \\ 0 & \text{if } adu_i \text{ is an opponent node} \end{cases} \\
fu_{ik} &= 1 \equiv adu_i \text{ has function label } k \\
at_{ij} &= 1 \equiv (i,j) \in E
\end{aligned}
$$

The local models described above provide us with four real-valued functions:

$$
\begin{aligned}
s_{cc} &: \{1,\ldots,n\} \mapsto \mathbb{R} \\
s_{ro} &: \{1,\ldots,n\} \mapsto \mathbb{R} \\
s_{fu} &: \{1,\ldots,n\} \times \{\nu_{cc}, \nu_a, \nu_s\}\mathbb{R} \\
s_{at} &: \{1,\ldots,n\}^2 \mapsto \mathbb{R}
\end{aligned}
$$

The objective function that we try to maximize is:

$$
\begin{aligned}
S_1 = &\sum_{i=1}^{n} s_{cc}(i) cc_i + \sum_{i=1}^{n} s_{ro}(i) ro_i \\
&+ \sum_{i=1}^{n}\sum_{k=1}^{3} s_{fu}(i,k) fu_{ik} + \sum_{i=1}^{n}\sum_{j=1}^{n} s_{at}(i,j) at_{ij}
\end{aligned}
$$

We refer to this objective function as S_1.

The constraints that we use can be split into different categories. First of all, the output structures need to respect the definitions related with the core variables. More specifically, there can be only one central claim:

$$\sum_{i=1}^{n} cc_i = 1 \tag{7.2}$$

All vertices have exactly one outgoing edge with the exception of central claim, which is a sink node:

$$\forall i \quad \left(cc_i + \sum_{j=1}^{n} at_{ij} \right) = 1 \tag{7.3}$$

All vertices have exactly one argumentative function:

$$\forall i \quad \sum_{k=1}^{3} fu_{ik} = 1 \tag{7.4}$$

The central claim must be a proponent node:

$$\forall i \quad cc_i \leq ro_i \tag{7.5}$$

This bans the case $cc_i = 1, ro_i = 0$, where the central claim is an opponent node. All other cases are allowed. The argumentative function should also match the central claim core variable:

$$\forall i \quad cc_i = fu_{i\nu_{cc}} \tag{7.6}$$

The next set of equations describe the relationship between argumentative functions and roles. A support edge can only occur between nodes of the same role, while attack edges only occur between nodes of different roles. We consider the edge from adu_i to adu_j. We build the following table:

at_{ij}	ro_i	$fu_{i\nu_s}$	ro_j	valid?	Comments
0	*	*	*	yes	No attachment, no restrictions
1	0	0	0	no	OPP attacks OPP
1	0	0	1	yes	OPP attacks PRO
1	0	1	0	yes	OPP supports OPP
1	0	1	1	no	OPP supports PRO
1	1	0	0	yes	PRO attacks OPP
1	1	0	1	no	PRO attacks PRO
1	1	1	0	no	PRO supports OPP
1	1	1	1	yes	PRO supports PRO

We now define $S_{ij} = ro_i + fu_{i\nu_s} + ro_j$. The table can be reduced to:

at_{ij}	S_{ij}	valid?
0	*	yes
1	0	no
1	1	yes
1	2	no
1	3	yes

We introduce a set of auxiliary variables, (psp_{ij}), which is set to 1 if and only if adu_i and adu_j form a "PRO supports PRO" pattern. in which case the ADUs need not to be attached and the defining constraint is as follows:

$$\forall i, j \quad 0 \leq S_{ij} - 3psp_{ij} \leq 2 \tag{7.7}$$

If $0 \leq S_{ij} \leq 2$, then psp_{ij} must be 0, or the sum will be negative. If $S_{ij} = 3$, then psp_{ij} must be 1, or the sum will be greater than 2. We now define $K_{ij} = S_{ij} - 2psp_{ij}$. The table can be completed:

at_{ij}	S_{ij}	psp_{ij}	K_{ij}	valid?
0	*	*	*	yes
1	0	0	0	no
1	1	0	1	yes
1	2	0	2	no
1	3	1	1	yes

If $at_{ij} = 1$, then the case is valid iff $K_{ij} = 1$. If $at_{ij} = 0$, then K_{ij} can take any value between 0 and 2. Therefore, we build the following constraint:

$$\forall i, j \quad at_{ij} \leq K_{ij} \leq 2 - at_{ij} \tag{7.8}$$

Other simpler constraints that we have used include the fact that there must be at least two proponent nodes in the graph $\sum_{i=1}^{n} ro_i \geq 2$. Also, the central claim must have at least one supporter. We introduce a set of binary variables, (scc_{ij}), which is set to 1 if and only if adu_i supports adu_j, and adu_j is the central claim. Given constraints 7.5, 7.7 and 7.8, we only need to check whether ADUs i and j are attached, and respectively proponent node and central claim. The structure of the constraint is similar to constraint 7.7:

$$\forall i, j \quad 0 \leq ro_i + at_{ij} + cc_j - 3scc_{ij} \leq 2 \tag{7.9}$$

The desired constraint follows from the previous definition:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} scc_{ij} \geq 1 \tag{7.10}$$

We also require that there be more proponent nodes than opponent nodes:

$$2 \sum_{i=1}^{n} ro_i \geq n \tag{7.11}$$

Finally we require that our graphs are acyclic.[11] We introduce an auxiliary set of integer variables, (c_i):

$$\forall i \quad 1 \leq c_i \leq n \tag{7.12}$$

$$\forall i, j \quad c_j \leq c_i - 1 + n(1 - at_{ij}) \tag{7.13}$$

[11] The proof of validity of this constraint can be found in appendix A

Replications of other ILP models We described in section 3.3 the approach of (Persing and Ng, 2016), also using ILP to build structures. We replicate their constraint set as closely as possible. Incidentally, as our corpus only contain pre-segmented single paragraphs, a fair number of their constraints are no longer necessary, and the rest match the ones we already implement. The difference resides in their objective function, which takes the following form once adapted to our local models:

$$
\begin{aligned}
\alpha &= 0.8 \\
p_X(\cdot) &= \frac{1}{1+e^{-s_X(\cdot)}} \text{ for } X \in \{cc, ro, fu, at\} \\
S_2 &= \sum_{i=1}^{n} \phi(p_{cc}(i), cc_i) + \sum_{i=1}^{n}\sum_{k=1}^{2} \phi(p_{fu}(i,k), fu_{ik}) \\
&\quad + \sum_{i=1}^{n}\sum_{j=1}^{n} \phi(p_{at}(i,j), at_{ij}) \\
\phi(x,y) &= 2\alpha(xy + (1-x)(1-y)) - (1-\alpha)(x(1-y) - (1-x)y)
\end{aligned}
$$

In our series of experiments we labeled the above objective function S_2.

7.2.3 Experiments and results

Evaluation procedure

In our experiments, we follow the setup of Peldszus and Stede (2015). We use the same train-test splits, resulting from 10 iterations of 5-fold cross validation, and adopt their evaluation procedure, where the correctness of predicted structures is assessed separately for the four subtasks, reported as macro averaged F1.

While these four scores cover important aspects of the structures, it would be nice to have a unified, summarizing metric for evaluating the decoded argumentation structures. To our knowledge, no such metric has yet been proposed, prior work just averaged over the different evaluation levels. Here, we will additionally report labeled attachment score (LAS) as a measure that combines attachment and the argumentative function labeling, as it is commonly used in dependency parsing. Note however, that this metric is not specifically sensitive for the importance of selecting the right central claim and also not sensitive for the dialectical dimension (choosing just one incorrect argumentative function might render the argumentative role assignment for the whole argumentative thread wrong).

For significance testing, we apply the Wilcoxon signed-rank test on the series of scores from the 50 train-test splits and assume a significance level of $\alpha = 0.01$.

	English				German			
model	cc	ro	fu	at	cc	ro	fu	at
Peldszus and Stede (2015)	.817	.750	.671	.663	**.849**	.755	.703	.679
Stab and Gurevych (2016)	.830		.745	.650				
base	.832	.762	.710	.690	.827	.757	.709	.696
base + BC	+.008	-.005	+.001	+.004	+.008	+.005	-.001	-.003
base + BC2		-.003	-.002	+.001	-.001	+.003		-.001
base + DR	+.005	+.018	+.019	+.003	+.002	-.002		-.001
base + VS	-.001	-.002	-.001	+.002	+.001		+.001	-.001
base + VEC	-.002	-.002	-.002	+.001	+.004	-.003	+.002	+.002
base + VLR		-.002		+.001	-.001		+.001	-.002
base + VST								-.001
base + SS				+.009				+.009
base + MC				+.012				+.016
all - VEC	**.840**	**.782**	.723	**.711**	.837	**.765**	.709	**.711**
all	.840	.780	.724	.710	.836	.762	**.712**	**.711**

Table 7.3: Evaluation scores for the base classifiers reported as macro avg. F1

Local models

The results of the experiment with the local models are shown in table 7.3. We first repeat the reported results of Peldszus and Stede (2015) and Stab and Gurevych (2016) for comparison. Below is our re-implementation of the classifiers of Peldszus and Stede (2015) (base), followed a feature analysis where we report on the impact of adding each new feature to the replicated baseline, reported as the delta.

Our replication of the baseline features (base) already provides a substantial improvement on all levels for the English version of the dataset. We attribute this mainly to the better performance of spaCy in parsing English. For German, the results are competitive. Only for central claim identification our replicated local models does not fully match the original model, which might be due to the fact that the spaCy parser does not offer a morphological analysis as deep as the mate parser and thus does not derive predictions for sentence mood.

Investigating the impact of the new features, the highest gain is achieved by adding the features for subordinate clauses (SS and MC) to the attachment classifier. Brown cluster unigrams give a moderate boost for central claim identification. Interestingly, the word-vector representation did not have a significant impact. The averaged word embeddings themselves (VEC) lowered the scores minimally for English and improved the results minimally for German, but increased the training time considerably. The distance measures based on word vectors (VST and VLR) yielded no improvement likewise.

	English				
model	cc	ro	fu	at	LAS
Peldszus and Stede (2015) (EG-equal)	.860	.721	.707	.692	.481
Stab and Gurevych (2016)	.857		.745	.683	
this work EG-equal	**.876**	.766	**.757**	**.722**	**.529**
this work ILP objective S_1	.864	.775	.749	**.722**	.523
this work ILP objective S_2	.869	**.783**	.740	.717	.519
this work ILP Persing & Ng	.869	.678	.732	.716	.491
	German				
model	cc	ro	fu	at	LAS
Peldszus and Stede (2015) (EG-equal)	**.879**	.737	.735	.712	.508
Stab and Gurevych (2016)					
this work EG-equal	.861	.730	.725	**.731**	.523
this work ILP objective S_1	.876	**.752**	**.740**	**.731**	**.526**
this work ILP objective S_2	.873	.743	.723	.729	.517
this work ILP Persing & Ng	.866	.634	.706	.723	.480

Table 7.4: Evaluation scores for the decoders reported as macro avg. F1 for the cc, ro, fu and at levels, and as labeled attachment score (LAS)

Taking all features together, excluding only the time-costly word embeddings (all - VEC), provides us with local models that achieve state of the art performance on all levels but **fu** for English and **cc** for German. We use this set of classifiers as the local models in all decoding experiments.

Global model

The results of the experiments with the decoders are shown in table 7.4. We again first repeat scores of prior studies and then present the results for the decoders introduced in section 7.2.2.

The overall best results for English are produced by the replication of the MST-based model of Peldszus and Stede (2015), followed by the novel ILP decoders. For German the novel ILP decoders score best, followed by the MST-based 'evidence graph' model. For both languages, there are no significant differences between these three models on any level.

The improvement in role, **LAS** and (for English only) **fu** of the MST method against the replication of Persing & Ng is statistically significant.

Chapter 8

Conclusion

8.1 Contributions

This thesis is the result of a prolonged effort to understand and recreate the organization of dialogue and argumentation.

While the structure of discourse has been the subject of decades of research, as discussed in chapter 2, the high-level structure of dialogue had been mostly overlooked. The creation of an annotated corpus dedicated to strategic dialogue was an incredible opportunity to advance the study of the particularities of multi-party dialogues. Parsing efforts were also mostly focused on monologue, which left a gap in the research for dialogue, where conventional methods proved inappropriate.

Regarding argumentation, the representation of persuasive text has drawn from logic and linguistics for a long time. While the multiple ways to express viewpoints have been explored in detail, the global structure of argumentation is still a unexplored field, especially for longer texts.

We present the following main contributions:

- We improved the formalization of the coherence of discourse, expanding its scope to multi-party dialogue;
- We designed a data extraction process for natural-language negotiations, providing evidence of the usefulness of discourse parsing;
- We evaluated the capabilities of tree-based methods in the production of the structure of dialogue;
- We created an efficient method to parse dialogue beyond trees, using more flexible and globally optimized graph structures;
- We annotated and compared the structures of discourse and argumentation on a corpus of persuasive texts;

- We applied our methods on the mostly-unexplored field of full-structure argumentative parsing.

Our new Right Frontier Constraint definition detailed in section 5.1 represents with greater semantic accuracy the interwoven threads present in multi-party chat dialogue. While group discussion is more chaotic than back-and-forth exchanges of two-party dialogue, and extremely distinct from the careful structure of monologue, we show that participants still abide by rules of coherence and respect the flow of conversation, even with access to the full history of the conversation granted by the medium of online chat.

Our process of identification of potentially hidden resources detailed in section 5.2 demonstrates an application of discourse parsing towards anaphora resolution. Local models of discourse structure prove useful in a context other than monologue, in situations where laconic responses do not carry any useful information when taken out of their reactive context.

We show in section 6.1 that we can successfully transfer the previous research on discourse parsing, applying tree-based methods to dialogue. Shallow features, adapted to the new domain, are verifiably reliable to model local discourse relations. We also build a solid case for the use of global optimization of decoded structures directly from the elementary discourse units, as opposed to bottom-up models implying a series of local decisions.

The ILP-based discourse parser presented in section 6.2 outperforms tree-based methods by a fair margin. We developed a more semantically accurate conversion of SDRT to dependency graphs, yielding an even wider margin compared to the MST algorithm. We show that we can predict directed acyclic graphs, introducing and formalizing original constraints for multi-party chat.

Finally, we transfer our methods on the field of argumentative parsing. We demonstrate again the efficiency of global optimization, and formalize the constraints of argumentative structure. We show that ILP-based methods work as well as tree-based methods on documents which actually *have* a tree structure.

8.2 Perspectives

Our research leaves some interesting questions unresolved, laying all the necessary groundwork for their resolution. We present three of them in this section.

CDU **prediction** We exposed in section 6.2.1 how we could remove CDUs from SDRT graphs, distributing discourse relations over the components of the CDU. We also mentioned there is no known method in the literature to predict clusters of units.

We see our progress on discourse parsing as an opportunity to tackle this problem. Distribution of CDUs typically create subsets of consecutive nodes with a high density of relations with the same label. Given a sufficiently accurate prediction of the discourse graphs of such small subsets, one could attempt to *recreate* CDUs by grouping EDUs sharing an antecedent (or target) *via* the same discourse relation.

One could also directly train a model to learn the parthood relation underlying CDUs, detecting units that belong to the same cluster.

Reliable detection of CDUs could open a new path towards interpretation of discourse and correct resolution of anaphoric links.

Comparative study of discourse and argumentation We described in section 7.1 a three-layer annotated corpus of short texts. While we performed a co-occurrence and common component analysis between argumentative and discourse structure, our descriptions scratched only the surface of the interaction between the two frameworks. In the same vein, we didn't apply our discourse parsing methods to the annotated discourse structure of the *Microtext* corpus, which would yield even more comparison material.

We believe the corpus can enable significant advances in the following open questions: to which extent discourse structures signal argumentative functions? Are argumentation and discourse structurally similar? Are the segmentations between discourse units and argumentative units aligned? Can the logical interpretation of discourse (in the SDRT framework) be tied to the formalization of arguments?

Study of long-distance attachments The *Settlers* corpus provides a number of long-distance attachments, where rhetorically connected units are separated by five or more EDUs, but short-distance attachments are far more frequent. As a result, the training dataset of our probabilistic model for pairs of EDUs is skewed in favor of the latter. We hypothesize this is the main reason the RFC didn't give us a significant increase in accuracy when used as an ILP constraint, as short-distance relations almost always follow the constraint.

We also observe that the performance of our parser degrades as the distance of attachment increases; a tendency also observed in the rest of the literature, intersentential relations being harder to predict than intra-sentential ones. Shallow features don't appear sufficient to predict long-distance attachment, and what exactly is needed in order to capture then is an open question. We hypothesize that deep semantic representation of utterances and background knowledge would play an important part in the matter, similarly to the general problem of implicit relation parsing.

Appendix A

Constraints

In this appendix, you will find the proofs related to the integer linear constraints used in sections 6.2.2 and 7.2.2.

Notation The constraints apply to a directed graph $G = \langle V, E, R \rangle$ with R being a function that provides labels for the edges in $E = (e_i)$. 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 $n = |V|$.

Per the definition of integer programming, all variables mentioned here take integer values. In addition, *binary* variables can only take the values 0 or 1 (this constraint will be implied whenever binary variable are introduced).

We define the n^2 binary variables a_{ij} and mn^2 binary variables r_{ijk}:

$$\begin{aligned} a_{ij} = 1 \quad &\equiv \quad (e_i, e_j) \in V \\ r_{ijk} = 1 \quad &\equiv \quad R(e_i, e_j) = k \end{aligned}$$

The two are tied by the unique label constraint:

$$\forall i, j \quad \left(\sum_k r_{ijk} \right) = a_{ij} \tag{A.1}$$

Acyclicity We require that our discourse and argumentative graphs are acyclic. We introduce an auxiliary set of integer variables, (c_i):

$$\forall i \quad 1 \leq c_i \leq n \tag{A.2}$$

$$\forall i, j \quad c_j \leq c_i - 1 + n(1 - a_{ij}) \tag{A.3}$$

If there is no edge between vertices e_i and e_j, then by definition $a_{ij} = 0$. In that case, inequality A.3 becomes $c_j \leq c_i - 1 + n$. Per inequality A.2, this always holds.

If there is an edge between vertices e_i and e_j, then by definition $a_{ij} = 1$ and inequality A.3 becomes $c_j \leq c_i - 1 \equiv c_j < c_i$.

Now assume, without loss of generality, that $(e_1, e_2..., e_k)$ is a chain. The constraint implies $(c_2 < c_1) \wedge (c_3 < c_2) \wedge \cdots \wedge (c_{k-1} < c_k) \equiv (c_1 < c_k)$. An extra edge from vertex e_k to e_1, forming a cycle, would imply $c_k < c_1$, which is incompatible with the previous result. Thus, constraints A.2 and A.3 enforce the acyclicity of the graph.

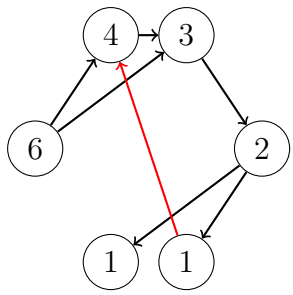


Figure A.1: Illustration of the acyclicity constraint.

Unique head We call *head* any vertex that has no incoming edge. We require our discourse graphs to have a unique head. We introduce an auxiliary set of binary variables, (h_i), and the following constraints:

$$\sum_i h_i = 1 \tag{A.4}$$

$$\forall j \quad 1 \leq nh_j + \sum_i a_{ij} \leq n \tag{A.5}$$

We show that $h_i = 1$ iff e_i is a head.

If e_j is a head, then $\forall i \quad a_{ij} = 0 \implies \sum_i a_{ij} = 0$. Per A.5, $1 \leq nh_j \implies h_j = 1$.

If e_j is *not* a head, then $\exists i \quad a_{ij} = 1 \implies \sum_i a_{ij} \geq 1$. Per A.5, $(nh_j + 1 \leq nh_j + \sum_i a_{ij} \leq n) \implies h_j = 0$. QED.

Equation A.4 trivially ensures the existence and uniqueness of the head.

Connectedness The combination of the acyclicity and unique head constraints is actually sufficent to ensure connectedness. To prove it, let $G = \langle V, E \rangle$ be an directed acyclic graph with a unique head. Let $G' = \langle V', E' \rangle$ be connected component of G. G' is acyclic as well, so we can define a partial order on V', where $u \leq v$ iff there is a directed path from u to v in G'. As V' is finite, it has at

least one minimal element. Moreover, any minimal element of V' is a head. Thus, any connected component of G contains a head.

As there must be only one head in G, this proves that G has only one connected component.

Unique sink node In argumentative graphs, the structural constraints are stronger: every vertex must have a single outgoing edge, except for a unique vertex, the *central claim*, which has none. We introduce an auxiliary set of binary variables, (cc_i), where $cc_i = 1$ iff e_i is the central claim (determined by other means). We introduce the following constraints:

$$\sum_{i=1}^{n} cc_i = 1 \tag{A.6}$$

$$\forall i \quad \left(cc_i + \sum_{j=1}^{n} a_{ij} \right) = 1 \tag{A.7}$$

Equation A.6 ensures the existence and uniqueness of a central claim. Let $e_i \in V$; if e_i is the central claim, then $cc_i = 1$ and equation A.7 becomes $\sum_{j=1}^{n} a_{ij} = 0 \implies \forall j \quad a_{ij} = 0$, which means that e_i has no outgoing edge, as intended. If e_i is *not* the central claim, then $cc_i = 0$ and equation A.7 becomes $\sum_{j=1}^{n} a_{ij} = 1 \implies \exists! i \quad a_{ij} = 1$, which corresponds to a unique outgoing edge, also as intended.

Associated with the acyclicity constraint, the unique sink constraint also ensures connectedness of the argumentative graph, with a proof similar to the previous paragraph. Additionally, equations A.6 and A.7 imply that $\sum_{i=1,j=1}^{n} a_{ij} = n - 1$, i.e. the graph has exactly $n - 1$ edges: the graph is thus a tree.

References

Stergos Afantenos and Nicholas Asher. 2010. Testing sdrt's right frontier. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1–9, Beijing, China, August. Coling 2010 Organizing Committee.

Stergos Afantenos, Nicholas Asher, Farah Benamara, Anaïs Cadilhac, Cédric Degremont, Pascal Denis, Markus Guhe, Simon Keizer, Alex Lascarides, Oliver Lemon, Philippe Muller, Soumya Paul, Vladimir Popescu, Verena Rieser, and Laure Vieu. 2012. Modelling strategic conversation: model, annotation design and corpus. In *Workshop on the Semantics and Pragmatics of Dialogue, Paris, France*. Université Paris 7, septembre.

Stergos Afantenos, Eric Kow, Nicholas Asher, and Jérémy Perret. 2015. Discourse parsing for multi-party 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.

Denis Apothéloz, Pierre-Yves Brandt, and Gustavo Quiroz. 1993. The function of negation in argumentation. *Journal of Pragmatics*, 19(1):23–38.

Nicholas Asher and Alex Lascarides. 1998. The semantics and pragmatics of presupposition. *Journal of Semantics*, 15(2):239–299.

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

N Asher and M Morreau. 1991. Common sense entailment: A modal theory of common-sense reasoning. In *Proc. 12th IJCAI*.

Nicholas Asher and Sylvain Pogodalla. 2010. Sdrt and continuation semantics. In *JSAI International Symposium on Artificial Intelligence*, pages 3–15. Springer.

Nicholas Asher, Daniel Hardt, and Joan Busquets. 1997. Discourse parallelism, scope, and ellipsis. In *Semantics and Linguistic Theory*, volume 7, pages 19–36.

Nicholas Asher, Antoine Venant, Philippe Muller, and Stergos Afantenos. 2011. Complex discourse units and their semantics. In Laurence Danlos Nicholas Asher, editor, *CID 2011 - Constraints in Discourse*, Agay, France, September.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the stac corpus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion

Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Nicholas Asher. 1986. Belief in discourse representation theory. *Journal of Philosophical Logic*, 15(2):127–189.

Nicholas Asher. 1993. Reference to abstract objects in english: a philosophical semantics for natural language metaphysics. *Studies in Linguistics and Philosophy. Kluwer, Dordrecht.*

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).*

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *journal of machine learning research*, 3(Feb):1137–1155.

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

Bernd Bohnet. 2010. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 89–97, Stroudsburg, PA, USA. Association for Computational Linguistics.

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.

Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

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.

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

Michael Collins and Nigel Duffy. 2002. New ranking algorithms for parsing and tagging: kernels over discrete structures, and the voted perceptron. In Isabelle Pierre, editor, *ACL '02 Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 263–270.

Koby Crammer and Yoram Singer. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of machine learning research*, 2(Dec):265–292.

Laurence Danlos. 2005. Comparing RST and SDRT Discourse Structures through Dependency Graphs. In *Proceedings of the Workshop on Constraints in Discourse (CID)*, Dortmund/Germany.

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)*.

Phan Minh Dung. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357.

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.

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

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.

Christiane Fellbaum. 1998. *WordNet*. Wiley Online Library.

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.

Vanessa Wei Feng and Graeme Hirst. 2014. A linear-time bottom-up discourse parser with constraints and post-editing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 511–521, Baltimore, Maryland, June. 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.

Eirini Florou, Stasinos Konstantopoulos, Antonis Koukourikos, and Pythagoras Karampiperis. 2013. Argument extraction for supporting public policy formulation. In *Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 49–54, Sofia, Bulgaria, August. Association for Computational Linguistics.

James B. Freeman. 1991. *Dialectics and the Macrostructure of Argument.* Foris, Berlin.

James B. Freeman. 2011. *Argument Structure: Representation and Theory.* Argumentation Library (18). Springer.

Gerald Gamrath, Tobias Fischer, Tristan Gally, Ambros M. Gleixner, Gregor Hendel, Thorsten Koch, Stephen J. Maher, Matthias Miltenberger, Benjamin Müller, Marc E. Pfetsch, Christian Puchert, Daniel Rehfeldt, Sebastian Schenker, Robert Schwarz, Felipe Serrano, Yuji Shinano, Stefan Vigerske, Dieter Weninger, Michael Winkler, Jonas T. Witt, and Jakob Witzig. 2016. The scip optimization suite 3.2. Technical Report 15-60, ZIB, Takustr.7, 14195 Berlin.

Jonathan Ginzburg. 2012. *The interactive stance.* Oxford University Press.

H Paul Grice. 1978. Further notes on logic and conversation. *1978*, 1:13–128.

J. Groenendijk and M. Stokhof. 1991. Dynamic predicate logic. *Linguistics and Philosophy*, 14:39–100.

Markus Guhe and Alex Lascarides. 2014. Game strategies in the settlers of catan. In *Proceedings of the IEEE Conference in Computational Intelligence in Games (CIG)*, Dortmund.

Charles Hamblin. 1987. *Imperatives.* Blackwells.

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.

J.R. Hobbs. 1979. Coherence and coreference. *Cognitive Science*, 3(1):67–90.

J. R. Hobbs. 1985. On the coherence and structure of discourse. Technical Report CSLI-85-37, Center for the Study of Language and Information, Stanford University.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal, September. Association for Computational Linguistics.

Julie Hunter, Nicholas Asher, Eric Kow, Jérémy Perret, and Stergos Afantenos. 2015. Defining the right frontier in multi-party dialogue. *SEMDIAL 2015 goDIAL*, page 95.

Shafiq Joty and Alessandro Moschitti. 2014. Discriminative reranking of discourse parses using tree kernels. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2049–2060, Doha, Qatar, October. Association for Computational Linguistics.

Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2012. A Novel Discriminative Framework for Sentence-Level Discourse Analysis. In *EMNLP-CoNLL*.

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*.

Dan Jurafsky and James H Martin. 2014. *Speech and language processing*. Pearson.

Hans Kamp, Josef Van Genabith, and Uwe Reyle. 2011. Discourse representation theory. In *Handbook of philosophical logic*, pages 125–394. Springer.

Hans Kamp. 1988. Discourse representation theory. *Natural Language at the computer*, pages 84–111.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289.

Mirella Lapata and Alex Lascarides. 2004. Inferring sentence-internal temporal relations. In *HLT-NAACL*, pages 153–160.

Alex Lascarides and Nicholas Asher. 1993. Temporal interpretation, discourse relations and commonsense entailment. *Linguistics and Philosophy*, 16(5):437–493.

Alex Lascarides and Nicholas Asher. 2009. Agreement, disputes and commitment in dialogue. *Journal of Semantics*, 26(2):109–158.

Huong Le Thanh, Geetha Abeysinghe, and Christian Huyck. 2004. Generating discourse structures for written text. In *Proceedings of Coling 2004*, pages 329–335, Geneva, Switzerland, Aug 23–Aug 27. COLING.

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.

Robert Malouf. 2002. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the Sixth Workshop on Natural Language Learning*, pages 49–55, Taipei, Taiwan.

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.

Daniel Marcu. 1997. The rhetorical parsing of natural language texts. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 96–103. Association for Computational Linguistics.

Daniel Marcu. 2000. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational linguistics*, 26(3):395–448.

James R. Martin. 1992. *English text: system and structure*. John Benjamins, Philadelphia/Amsterdam.

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 McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *11th Conference of the European Chapter of the 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*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Eleni Miltsakaki, Rashmi Prasad, Aravind K Joshi, and Bonnie L Webber. 2004. The Penn Discourse Treebank. In *LREC*.

Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. 2007. Automatic detection of arguments in legal texts. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, pages 225–230.

Richard Montague, Bruce Vermazen, and Richmond H Thomason. 1976. Formal philosophy: Selected papers of richard montague.

P. Muller, M. Vergez, L. Prevot, N. Asher, F. Benamara, M. Bras, A. Le Draoulec, and L. Vieu. 2012a. Manuel d'annotation en relations de discours du projet Annodis. *Carnets de Grammaire*, 21.

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

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT*. Citeseer.

Stephan Oepen, Dan Flickinger, Kristina Toutanova, and Christopher D Manning. 2004. Lingo redwoods. *Research on Language and Computation*, 2(4):575–596.

M. J. Osborne and A. Rubinstein. 1994. *A Course in Game Theory*. MIT Press.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Andreas Peldszus and Manfred Stede. 2013. From argument diagrams to automatic argument mining: A survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 7(1):1–31.

Andreas Peldszus and Manfred Stede. 2015. Joint prediction in mst-style discourse parsing for argumentation mining. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 938–948, Lisbon, Portugal, September. Association for Computational Linguistics.

Andreas Peldszus and Manfred Stede. 2016. An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon 2015 / Vol. 2*, pages 801–815, London. College Publications.

Andreas Peldszus. 2014. Towards segment-based recognition of argumentation structure in short texts. In *Proceedings of the First Workshop on Argumentation Mining*, pages 88–97, Baltimore, Maryland, June. Association for Computational Linguistics.

Jérémy Perret, Stergos Afantenos, Nicholas Asher, and Alex Lascarides. 2014. Revealing resources in strategic contexts. *SEMDIAL 2015 DialWatt*.

Jérémy Perret, Stergos Afantenos, Nicholas Asher, and Mathieu Morey. 2016. Integer linear programming for discourse parsing. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 99–109, San Diego, California, June. Association for Computational Linguistics.

Isaac Persing and Vincent Ng. 2016. End-to-end argumentation mining in student essays. In *Proceedings of the 2016 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, pages 1384–1394, San Diego, California, June. Association for Computational Linguistics.

Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. 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: Volume 2 - Volume 2*, ACL '09, pages 683–691, Stroudsburg, PA, USA. Association for Computational Linguistics.

Massimo Poesio and David R Traum. 1997. Conversational actions and discourse situations. *Computational intelligence*, 13(3):309–347.

Livia Polanyi and Remko Scha. 1984. A syntactic approach to discourse semantics. In *Proceedings of the 10th International Conference on Computational Linguistics (COLING84)*, pages 413–419, Stanford.

Livia Polanyi. 1985. A theory of discourse structure and discourse coherence. In P. D. Kroeber W. H. Eilfort and K. L. Peterson, editors, *Papers from the General Session at the 21st Regional Meeting of the Chicago Linguistic Society*. Chicago Linguistic Society.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proc. of the 6th International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

Laurent Prévot and Laure Vieu. 2008. The moving right frontier. *Pragmatics and beyond new series*, 172:53.

Attapol Rutherford and Nianwen Xue. 2014. Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. In *EACL*, volume 645, page 2014.

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.

Manami Saito, Kazuhide Yamamoto, and Satoshi Sekine. 2006. Using phrasal patterns to identify discourse relations. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, NAACL-Short '06, pages 133–136, Stroudsburg, PA, USA. Association for Computational Linguistics.

Emanuel A Schegloff. 2007. *Sequence organization in interaction: Volume 1: A primer in conversation analysis*, volume 1. Cambridge University Press.

David Schlangen. 2003. *A coherence-based approach to the interpretation of non-sentential utterances in dialogue*. Ph.D. thesis, University of Edinburgh. College of Science and Engineering. School of Informatics.

Natalie Schluter. 2014. On maximum spanning dag algorithms for semantic dag parsing. In *Proceedings of the ACL 2014 Workshop on Semantic Parsing*, pages 61–65, Baltimore, MD, June. Association for Computational Linguistics.

Noah A. Smith. 2011. *Linguistic Structure Prediction*. Synthesis Lectures on Human Language Technologies. Morgan and Claypool, May.

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.

Christian Stab and Iryna Gurevych. 2014a. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014b. Identifying argumentative discourse structures in persuasive essays. In *EMNLP*, pages 46–56.

Christian Stab and Iryna Gurevych. 2016. Parsing Argumentation Structures in Persuasive Essays. *ArXiv e-prints*, April. https://arxiv.org/abs/1604.07370.

Manfred Stede and Arne Neumann. 2014. Potsdam commentary corpus 2.0: Annotation for discourse research. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 925–929, Reikjavik.

Manfred Stede, Stergos Afantenos, Andreas Peldszus, Nicholas Asher, and Jérémy Perret. 2016. Parallel Discourse Annotations on a Corpus of Short Texts. In Nicoletta Calzolari et al., editor, *Proc. of the Ninth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia.

Manfred Stede, editor. 2016. *Handbuch Textannotation: Potsdamer Kommentarkorpus 2.0*. Universitätsverlag Potsdam. Available online: http://nbn-resolving.de/urn:nbn:de:kobv:517-opus4-82761.

Philip J Stone, Dexter C Dunphy, and Marshall S Smith. 1966. The general inquirer: A computer approach to content analysis.

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.

Rajen Subba, Barbara Di Eugenio, and Elena Terenzi. 2006. Building lexical resources for princpar, a large coverage parser that generates principled semantic representations. In *LREC06, the fifth International Conference on Language Resources and Evaluation*, pages 327–332.

Charles Sutton, Andrew McCallum, and Khashayar Rohanimanesh. 2007. Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. *Journal of Machine Learning Research*, 8(Mar):693–723.

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

Klaus Teuber. 1995. *Die Siedler von Catan: Regelheft.* Kosmos Verlag, Stuttgart, Germany.

Stephen Toulmin. 1958. *The uses of argument.* Cambridge: Cambridge University Press.

Frans H Van Eemeren and Rob Grootendorst. 1992. *Argumentation, communication, and fallacies: A pragma-dialectical perspective.* Lawrence Erlbaum Associates, Inc.

Vladimir N Vapnik. 1995. The nature of statistical learning theory.

Antoine Venant, Nicholas Asher, Philippe Muller, Pascal Denis, and Stergos Afantenos. 2013. Expressivity and comparison of models of discourse structure. In *Proceedings of the SIGDIAL 2013 Conference*, pages 2–11, Metz, France, August. Association for Computational Linguistics.

Bonnie Webber. 1988. Discourse deixis: Reference to discourse segments. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics*, pages 113–122. Association for Computational Linguistics, Morristown, NJ.

Ben Wellner and James Pustejovsky. 2007. Automatically Identifying the Arguments of Discourse Connectives. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 92–101, Prague, Czech Republic, June. Association for Computational Linguistics.

Yoam Sholam and Kevin Leyton-Brown. 2009. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations.* Cambridge University Press.

Zhi-Min Zhou, Yu Xu, Zheng-Yu Niu, Man Lan, Jian Su, and Chew Lim Tan. 2010. Predicting discourse connectives for implicit discourse relation recognition. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1507–1514. Association for Computational Linguistics.