# Chapter 7

# Discourse-Aware Sentiment Analysis

Although coarse-grained sentiment analysis methods presented in the previous chapter do a fairly good job at classifying the overall polarity of a message, a crucial limitation of all these systems is that they completely overlook the structural nature of their input by either considering it as a single whole (*e.g.*, bag-of-features approaches) or analyzing it as a monotone sequence of equally important elements (*e.g.*, recurrent neural methods). Unfortunately, both of these solutions violate the hierarchical principle of language (de Saussure and Engler, 1990; Hjelmslev, 1970), which states that complex linguistic units are formed from smaller language elements in the bottom-up way: *e.g.*, words are created by putting together morphemes, sentences are made of several words, and discourse is composed of multiple coherent sentences. Moreover, apart from this inherent structural heterogeneity, even units of the same linguistic level might play a different role and be of unequal importance when joined syntagmatically into the higher-level whole: for example, in words, the root morpheme typically conveys more lexical meaning than the affixes; in sentences, the syntactic head usually dominates its grammatical dependents; and, in discourse, one of the sentences frequently expresses more relevant ideas than the rest of the text.

Exactly the lack of discourse information was one of the main reasons for the misclassifications made by the systems of Severyn and Moschitti (2015b), Baziotis et al. (2017), and our own LBA method in Examples 6.5.3, 6.5.4, and 6.5.5. Since none of these approaches explicitly took discourse structure into account, we decided to check whether making the last of these solutions (the LBA classifier) aware of discourse phenomena would improve its results. But before we present our experiments, we first would like to make a short digression into the theory of discourse and give an overview of the most popular approaches to text-level analysis that exist in the literature nowadays. Afterwards, in Section 7.2, we will describe the way how we inferred discourse information for the PotTS and SB10k data. Then, in Section 7.3, we will summarize the current state of the art in discourse-aware sentiment analysis (DASA) and also present our own methods, evaluating them on the aforementioned

datasets. After analyzing the effects of various common factors (such as the impact of the underlying sentiment classifier and the amenability of various discourse relation schemes to different DASA approaches), we will recap our results and summarize our findings in the last part of this chapter.

## 7.1 Discourse Analysis

Since the main focus of our experiments will be on *discourse analysis*, we first need to clarify what discourse analysis actually means and which common ways there are to represent and analyze discourse automatically.

In a nutshell, discourse analysis is an area of research which explores and analyzes language phenomena beyond the sentence level (Stede, 2011). Although the scope of this research can be quite large, ranging from the use of pronouns in a sentence to the logical composition of the whole document, in our work we will primarily concentrate on the coherence structure of a text, *i.e.*, its segmentation into *elementary discourse units* (typically single propositions) and induction of hierarchical *coherence relations* (semantic or pragmatic links) between these EDUs.

Although the idea of splitting the text into smaller meaningful pieces and inferring semantic relationships between these parts is anything but new, dating back to the very origins of general linguistics (Aristotle, 2010) and in particular its structuralism branch (de Saussure and Engler, 1990), an especially big surge of interest in this field happened in the 1970-s with the fundamental works of van Dijk (1972) and van Dijk and Kintsch (1983), who introduced the notion of local and global coherence, defining the former as a set of "rules and conditions for the well-formed concatenation of pairs of sentences in a linearly ordered sequence" and specifying the latter as constraints on the macro-structure of the narrative (see Hoey, 1983). Similar ideas were also proposed by Longacre (1979, 1996), who considered the paragraph as a unit of tagmemic grammar that was composed of multiple sentences according to a predefined set of compositional principles. Almost contemporary with these works, Winter (1977) presented an extensive study of various lexical means which could connect two sentences and grouped these means into two major categories: MATCHING and LOGICAL SEQUENCE, depending on whether they introduced sentences that were giving more details on the preceding content (MATCHING) or adding new information to the narrative (LOGICAL SEQUENCE).

The increased interest of traditional linguistics in text-level analysis has rapidly spurred the attention of the broader NLP community. Among the first who stressed the importance of discourse phenomena for automatic generation and understanding of texts was Hobbs (1979),

who argued that semantic ties between sentences were one the most important component for building a coherent discourse. Similarly to Winter, Hobbs also proposed a classification of inter-sentence relations, dividing them into ELABORATION, PARALLEL, and CONTRAST. Albeit this taxonomy was obviously too small to accommodate all possible semantic and pragmatic relationships that could exist between two clauses, this division had laid the foundations for many successful approaches to automatic discourse analysis that appeared in the following decades.

**RST.** One of the best-known such approaches—*Rhetorical Structure Theory* or *RST*—was presented by Mann and Thompson (1988). Besides revising Hobbs' inventory of discourse relations and expanding it to 23 elements (including new items such as ANTITHESIS, CIRCUMSTANCE, EVIDENCE, and ELABORATION), the authors also grouped all coherence links into nucleus-satellite (hypotactic) and multinuclear (paratactic) ones, depending on whether the arguments of these edges were of different or equal importance to the content of the whole text. Based on this grouping, they formally described each relation as a set of constraints on the *Nucleus* (N), *Satellite* (S), *the N+S combination*, and *the effect* of the whole combination on the reader (R). An excerpt from the original description of the ANTITHESIS relation is given in Example 7.1.1

**Example 7.1.1 (Definition of the ANTITHESIS Relation)**
***Relation Name:*** ANTITHESIS

***Constraints on N:*** *W has positive regard for the situation presented in N*

***Constraints on S:*** *None*

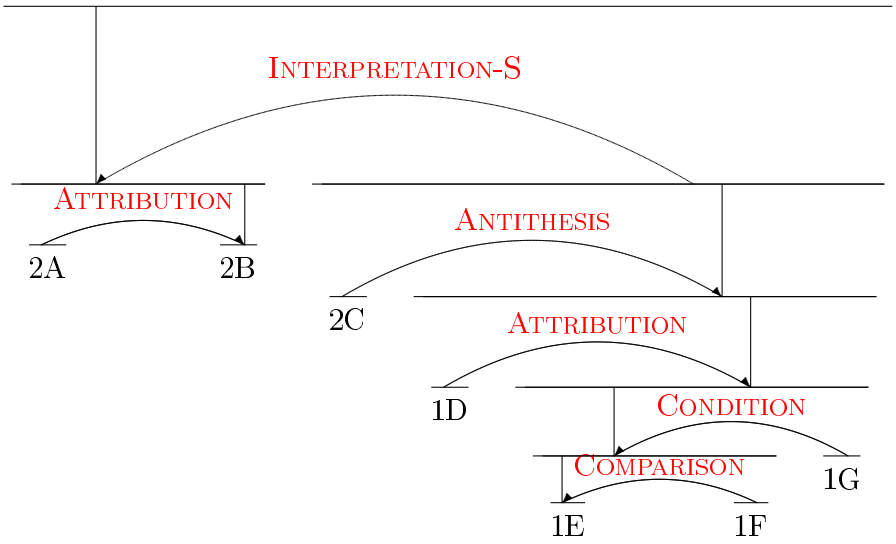
***Constraints on the N+S Combination:*** *the situations presented in N and S are in contrast (i.e., are (a) comprehended as the same in many respects, (b) comprehended as differing in a few respects and (c) compared with respect to one or more of these differences ); because of an incompatibility that arises from the contrast, one cannot have positive regard for both the situations presented in N and S; comprehending S and the incompatibility between the situations presented in N and S increases R's positive regard for the situation presented in N*

***Effect:*** *R's positive regard for N is increased*

***Locus of the Effect:*** *N*

The authors then defined the general structure of discourse as a projective (constituency) tree whose nodes were either elementary discourse units or subtrees which were connected to each other via discourse relations.

You can see an example of such discourse tree from the original Rhetorical Structure Treebank (Carlson et al., 2001) in Figure 7.1.




[Analysts said,]$^{1A}$ [profit for the dozen or so big drug makers, as a group, is estimated to have climbed between 11% and 14%.]$^{1B}$ [While that's not spectacular,]$^{1C}$ [Neil Sweig, an analyst with Prudential Bache, said]$^{1D}$ [that the rate of growth will "look especially good]$^{1E}$ [as compared to other companies]$^{1F}$ [if the economy turns downward."]$^{1G}$ (WSJ-2341; Carlson et al., 2001)

*Figure 7.1: Example of an RST-Tree*

Despite its immense popularity and practical utility (see Marcu, 1998; Yoshida et al., 2014; Bhatia et al., 2015; Goyal and Eisenstein, 2016), RST has often been criticized for the rigidness of the imposed tree structure (Wolf and Gibson, 2005) and arbitrariness of the distinguished discourse links (Nicholas, 1994; Miltsakaki et al., 2004a). As a result of this criticism, two alternative approaches to automatic discourse analysis were proposed in later works.

**PDTB.** One of these approaches—PDTB (named so after the Penn Discourse Treebank [Prasad et al., 2004])—was developed by the research group at the University of Pennsylvania (Miltsakaki et al., 2004a,b; Prasad et al., 2008) and at its core represents an *underspecification of RST*, where instead of fully specifying the hierarchical structure of the whole text and providing an all-embracing set of discourse relations, the authors mainly focused on the grammatical and lexical means that could connect two sentences (*connectives*), expressing a semantic relationship (*sense*) between these predicates. Typical such means are coordinating or subordinating conjunctions (*e.g.*, *and*, *because*, *since*) and discourse adverbials (*e.g.*, *however*, *otherwise*, *as a result*), which can denote a CONJUNCTION, a COMPARISON, a CONTRAST, or some other sense between two sentential arguments (ARG1 and ARG2).

Apart from *explicitly* mentioned connectives, Prasad et al. (2004) also allowed for situa-

tions where connectives were missing but can be easily inferred from the text. They called these cases *implicit* discourse relations and demanded the arguments of such structures be determined as well. Furthermore, if there was no connective at all, the authors of PDTB distinguished three different possibilities:

- the coherence relation was either expressed by an alternative lexical means which made the connective redundant (AltLex),
- or it was achieved by referring to the same entities in both arguments (EntRel),
- or there was no coherence relation at all (NoRel);

and also provided a special Attribution label for marking the authors of reported speech.

Example 7.1.2 shows the previous fragment of the Rhetorical Treebank now annotated according to the PDTB scheme. As we can see from the analysis, PDTB is indeed more flexible than RST, as it allows its discourse units (arguments) to overlap, be disjoint or even embedded into other segments. The assignment of sense relations is also more straightforward and mainly determined by the connectives that link the arguments. But, at the same time, the structure of this annotation is completely flat so that we can neither infer which of the sentences plays a more prominent role nor see the modification scope of other supplementary statements.

**Example 7.1.2 (Example of PDTB Analysis)**

*Analysts said,* [*profit for the dozen or so big drug makers, as a group, is estimated to have climbed between 11% and 14%.*]$_{rel1:arg1}$ [IMPLICIT:=*in fact*]$_{rel1:connective}$ [[EXPLICIT:=*While*]$_{rel2:connective}$ [*that's not spectacular*]$_{rel2:arg2}$]$_{rel1:arg2}$;
*Neil Sweig, an analyst with Prudential Bache, said* [[[*that the rate of growth will "look especially good as compared to other companies*]$_{rel3:arg1}$ [EXPLICIT:*if*]$_{rel3:connective}$[*the economy turns downward*]$_{rel3:arg2}$]$_{rel2:arg1}$]$_{rel1:arg2}$."

**SDRT.** Another alternative to RST—Segmented Discourse Representation Theory or SDRT—was proposed by Lascarides and Asher (2001). Although developed from a completely different angle of view (the SDRT authors mainly drew their inspiration from predicate logic, dynamic semantics, and anaphora theory), this theory shares many of its features with the standard rhetorical structure, as it also assumes a graph-like structure of text and distinguishes between coordinating and subordinating relations. However, unlike RST, segmented discourse representation explicitly allows the text structure to be any graph and not only tree (*i.e.*, a discourse node can have multiple parents and can also be connected via multiple links to the same vertex), provided that it does not have crossing dependencies

(*i.e.*, does not violate the right-frontier constraint). In this respect, SDRT can be viewed as a *structural generalization of RST*.

We can also notice the relatedness of the two approaches by looking at the SDRT analysis of the previous RST fragment in Example 7.2. Although the names of the relations in the presented graph differ from those used in RST, many of these links have the same (or at least similar) meaning as the respective edges in the first analysis: for example, the SOURCE relation in SDRT almost completely corresponds to the ATTRIBUTION edge in Example 7.1, and the CONTRAST link is similar to the COMPARISON relation defined by Mann and Thompson (1988).

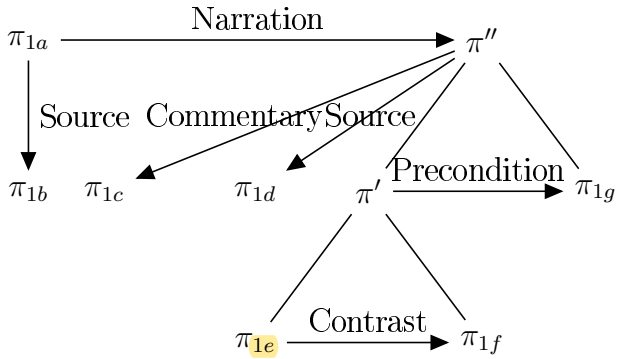


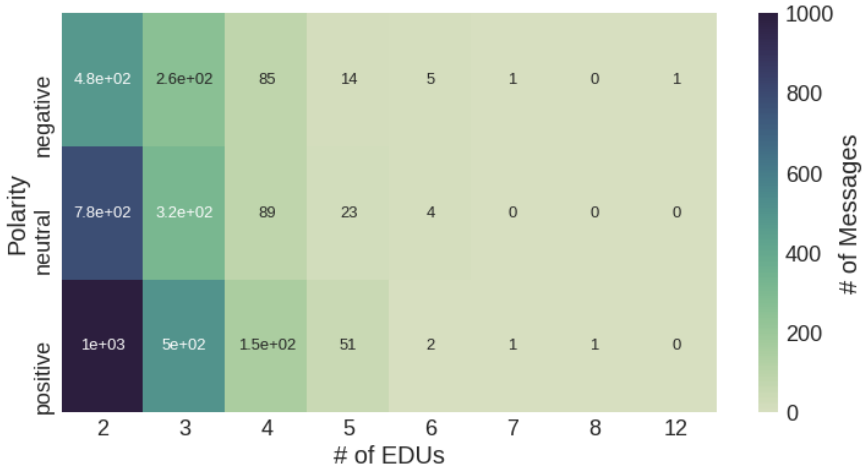

*Figure 7.2: Example of an SDRT graph*

**Final choice.** Because it was unclear which of these approaches (RST, PDTB, or SDRT) would be more amenable to our sentiment experiments, we have made our decision by considering the following theoretical and practical aspects: From theoretical perspective, we wanted to have a strictly hierarchical discourse structure for each analyzed tweet so that we could infer the semantic orientation of that message by recursively accumulating polarity scores of its elementary discourse segments. From practical point of view, since there was no discourse parser readily available for German, we wanted to have a maximal assortment of such systems available for English so that we could pick one that would be easiest to retrain on German data. Fortunately, both of these concerns have lead us to the same solution—Rhetorical Structure Theory, which was the only formalism which explicitly guaranteed a single root for each analyzed text and also offered a wide variety of open-source parsing systems (*e.g.*, Hernault et al., 2010; Feng and Hirst, 2014; Ji and Eisenstein, 2014; Yoshida et al., 2014; Joty et al., 2015).

## 7.2 Data Preparation

To prepare the data for our experiments, we split all microblogs from the PotTS and SB10k corpora into elementary discourse units using the ML-based discourse segmenter of Sidarenka

et al. (2015b). After filtering out all tweets that had at most one EDU,[1] we obtained 4,771 messages (12,137 segments) for PotTS and 3,763 posts (9,625 segments) for the SB10k corpus. In the next step, we assigned polarity scores to the segments of these microblogs with the help of our lexicon-based attention classifier, analyzing each elementary unit in isolation, independently of the rest of the tweet. We again used the same 70-10-20 split into training, development, and test sets as we did in our previous chapters.




*(a) PotTS*


*(b) SB10k*

*Figure 7.3: Distribution of elementary discourse units and polarity classes in the training and development sets of PotTS and SB10k*

As we can see from the statistics in Figure 7.3, most tweets which consist of multiple EDUs typically have two or three segments, whereas messages with more than three discourse units are extremely rare. This is also not surprising regarding that the maximum length of a microblog is constrained to 140 characters. Nonetheless, even with this severe length restriction, there still are a few messages which have up to 13 EDUs. Since it was somewhat

[1]Since the focus of this chapter is mainly on discourse phenomena, we skip all messages which consist of a single discourse segment, because their overall polarity is unaffected by the discourse structure and can be normally determined with the standard discourse-unaware sentiment techniques.

unexpected for us to see that many segments in a single tweet, we decided to have a closer look at these cases. As it turned out, such high number of discourse units typically resulted from spurious punctuation marks, which were carelessly used by Twitter users and evidently confused our segmenter (see Example 7.2.1).

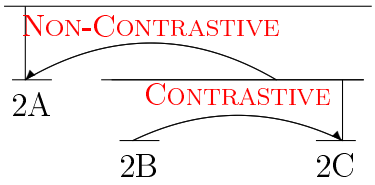
**Example 7.2.1 (SB10k Tweet with 13 EDUs)**
**Tweet:** [Guinness on Wheelchairs :]$_1$ [Das .]$_2$ [Ist .]$_3$ [Verdammt .]$_4$ [Noch .]$_5$ [Mal .]$_6$ [Einer .]$_7$ [Der .]$_8$ [Besten .]$_9$ [Werbespots .]$_{10}$ [Des .]$_{11}$ [Jahrzehnts .]$_{12}$ [( Auch ...]$_{13}$
[*Guinness on Wheelchairs :*]$_1$ [*This .*]$_2$ [*Is .*]$_3$ [*Gosh .*]$_4$ [*Darn .*]$_5$ [*It .*]$_6$ [*One .*]$_7$ [*Of .*]$_8$ [*The best .*]$_9$ [*Commercials .*]$_{10}$ [*Of .*]$_{11}$ [*The Decade .*]$_{12}$ [*( Also ...*]$_{13}$

Another noticeable trend that we can see in the data is that the distribution of polar classes in messages with multiple segments largely corresponds to the frequencies of these polarities in the complete datasets: For example, the positive semantic orientation still dominates the PotTS corpus, whereas the neutral polarity constitutes the vast majority of the SB10k set. At the same time, negative microblogs again are the least represented class in both cases and account for only 22% of the former corpus and for 16% of the latter data.

To obtain RST trees for these messages, we retrained the DPLP discourse parser of Ji and Eisenstein (2014) on the Potsdam Commentary Corpus (PCC 2.0; Stede and Neumann, 2014), after converting all discourse relations of this dataset to the binary scheme {CONTRASTIVE, NON-CONTRASTIVE} as suggested by Bhatia et al. (2015).[2] In contrast to the original DPLP implementation though, we did not use Brown clusters (Brown et al., 1992), because this resource was not available for German, and did not apply the linear projection of the features, because the released parser code was missing this component either. In part due to these modifications, but mostly because of the specifics of the German language (richer morphology, higher lexical variety, and syntactic ambiguity), the results of the retrained model were considerably lower than the figures reported for the English treebank, amounting to 0.777, 0.512, and 0.396 $F_1$ for span, nuclearity, and relation classification on PCC 2.0 versus corresponding 82.08, 71.13, and 61.63 $F_1$ on the RST Treebank.[3]

An example of an automatically induced RST tree for a Twitter message is shown in Figure 7.4. As we can see from this picture, the adapted parser can correctly distinguish between contrastive and non-contrastive relations (even though it only predicts the former class for two percent of all edges on the PotTS and SB10k data [see Figure 7.5]), but apparently struggles with the disambiguation of the nuclearity status, assigning the highest importance

[2] See Table 7.3 for more details regarding this mapping.
[3] Following Ji and Eisenstein (2014), we use the span-based evaluation metric of Marcu (2000).




[Mooooiiinn.]$^{2A}$ [Gegen solche Nächte hilft die beste Kur nicht.]$^{2B}$ [Aber Kaffee!]$^{2C}$ (PotTS; Sidarenka, 2016)
[Helllloooo!]$^{2A}$ [Even the best cure won't help against such nights.]$^{2B}$ [But coffee!]$^{2C}$

*Figure 7.4: Example of an automatically constructed RST-tree for a Twitter message*

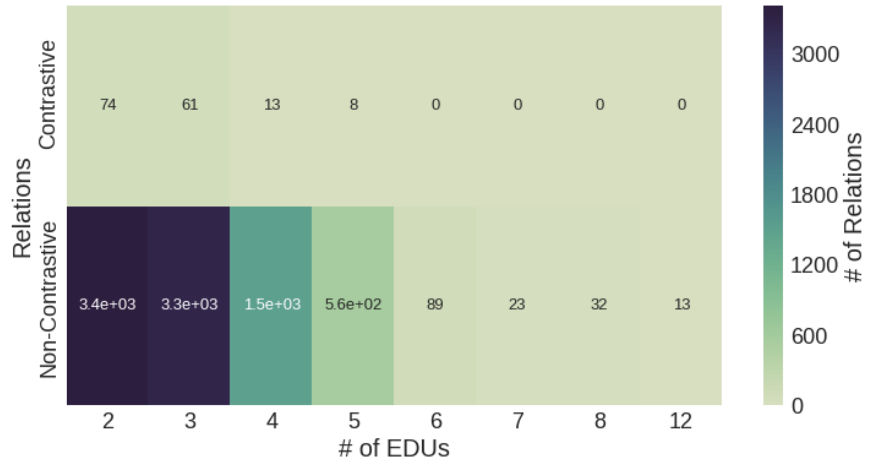
in this example to the initial discourse segment ("Mooooiiinn." [*Helllloooo!*]), which is merely a greeting, and weighing the second EDU ("Gegen solche Nächte hilft die beste Kur nicht." [*Even the best cure won't help against such nights.*]) less than the third one ("Aber Kaffee!" [*But coffee!*]), although traditional RST would rather consider both units as equally relevant and join them via the multi-nuclear CONTRAST link.

## 7.3 Discourse-Aware Sentiment Analysis

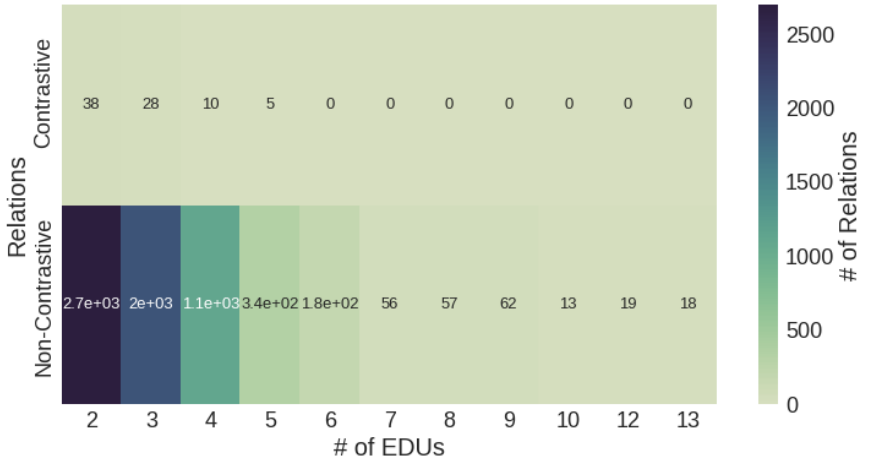
Now, before we use these data in our sentiment experiments, let us first revise the most prominent approaches to discourse-aware sentiment analysis that exist in the literature nowadays.

**Example 7.3.1 (Polarity reversal via discourse antithesis)**
This film should be brilliant. It sounds like a great plot, the actors are first grade, and the supporting cast is good as well, and Stallone is attempting to deliver a good performance. However, it can't hold up. (Pang et al., 2002)

As it turns out, even the very first works on opinion mining already pointed out the importance of discourse phenomena for the classification of the overall polarity of a text. For example, in the seminal paper of Pang et al. (2002), where the authors tried to predict the semantic orientation of movie reviews, they quickly realized the fact that it was insufficient to rely on the mere presence or even the majority of polarity clues in text, because these clues could any time be reversed by a single counter-argument of the critic (see Example 7.3.1). This observation was also confirmed by Polanyi and Zaenen (2006), who ranked discourse relations among the most important factors which could significantly affect the intensity and polarity of a sentiment. To prove this claim, they gave several convincing examples, where a concessive statement considerably weakened the strength of a polar opinion, and vice versa, an elaboration notably increased its persuasiveness.




*(a) PotTS*




*(b) SB10k*

*Figure 7.5: Distribution of discourse relations in the training and development sets of PotTS and SB10k*

Pang and Lee (2004) were also among the first who incorporated a discourse-aware component into a document-level sentiment classifier. In an attempt to improve the classification accuracy on IMDB, they presented a two-stage system in which the first predictor distinguished between subjective and objective statements by constructing a graph of all sentences (linking each sentence to its neighbors and also connecting it to two abstract polarity nodes) and then partitioning this graph into two clusters (subjective and objective) based on its minimum cut; the second classifier then inferred the overall polarity of the text by only looking at the sentences from the first (subjective) group. With this method, Pang and Lee achieved a statistically significant improvement (86.2% versus 85.2% for the Naïve Bayes system and 86.15% versus 85.45% for SVM) over classifiers which analyzed all sentences of the review at once, without any filtering.

Although an oversimplification, the core idea that locally adjacent sentences were likely to share the same subjective orientation (*local coherence*) was dominating the following DASA

research for almost a decade. For example, Riloff et al. (2003) also improved the accuracy of their Naïve Bayes predictor of subjective expressions by almost two percent after adding a set of local coherence features. Similarly, Hu and Liu (2004) could better disambiguate users' attitudes to particular product attributes by taking the semantic orientation of previous sentences into account.

At the same time, another line of discourse-aware sentiment research concentrated on the joint classification of all opinions in text, where in addition to predicting each sentiment in isolation, the authors also sought to maximize the "total happiness" (*global coherence*) of these assignments, ensuring that related subjective statements received agreeing polarity scores. Notable works in this direction were done by Snyder and Barzilay (2007), who proposed the Good Grief algorithm for predicting users' satisfaction with different restaurant aspects, and Somasundaran et al. (2008a,b), who introduced the concept of *opinion frames* (OF)—a special data structure for capturing the relations between opinions in discourse. Depending on the type of these opinions (arguing [$A$] or sentiment [$S$]), their polarity towards the target (positive [$P$] or negative [$N$]), and semantic relationship between these targets (alternative [*Alt*] or the same [*same*]), the authors distinguished 32 types of possible frames: *SPSPsame*, *SPSNsame*, *APAPalt*, etc.; dividing them into reinforcing and non-reinforcing ones. In later works, Somasundaran et al. (2009b,a) also presented two joint inference frameworks (based on the iterative classification and integer linear programming) for determining the best configuration of all frames in text, achieving 77.72% accuracy on frame prediction in the AMI meeting corpus (Carletta et al., 2005).

An attempt to unite local and global coherence was made by McDonald et al. (2007), who tried to simultaneously predict the polarity of a document and classify the semantic orientations of all its sentences. For this purpose, the authors devised an undirected probabilistic graphical model based on the structured linear classifier (Collins, 2002). Similarly to Pang and Lee (2004), they connected the label nodes of each sentence to the labels of its neighboring clauses and also linked these nodes to the overarching vertex representing the polarity of the text. After optimizing this model with the MIRA learning algorithm (Crammer and Singer, 2003), McDonald et al. achieved an accuracy of 82.2% for document-level classification and 62.6% for sentence-level prediction on a corpus of online product reviews, outperforming pure document and sentence classifiers by up to four percent. A crucial limitation of this system though was that its optimization required the gold labels of sentences and documents to be known at training time, which considerably limited its applicability to other domains with no such data.

Another significant drawback of all previous approaches is that they completely ignored traditional discourse theory and, as a result, severely oversimplified discourse structure. Among the first who tried to overcome this omission were Voll and Taboada (2007), who

proposed two discourse-aware enhancements of their lexicon-based sentiment calculator (SO-CAL). In the first method, the authors let the SO-CAL analyze only the topmost nucleus EDU of each sentence, whereas in the second approach, they expanded its input to all clauses that another classifier had considered as relevant to the main topic of the document. Unfortunately, the former solution did not work out as well as expected, yielding 69% accuracy on the corpus of Epinion reviews (Taboada et al., 2006), but the latter system could perform much better, achieving 73% on this two-class prediction task.

Other ways of adding discourse information to a sentiment system were explored by Heerschop et al. (2011), who experimented with three different approaches: (i) increasing the polarity scores of words which appeared near the end of the document, (ii) assigning higher weights to the nucleus tokens, and finally (iii) learning separate scores for nuclei and satellites using a genetic algorithm. An evaluation of these methods on the movie review corpus of Pang and Lee (2004) showed better performance of the first option (0.608 accuracy and 0.597 macro-$F_1$), but the authors could significantly improve the results of the last classifier at the end by adding an offset to the decision boundary of this method, which increased both its accuracy and macro-averaged $F_1$ to 0.72.

Further notable contributions to RST-based sentiment analysis were made by Zhou et al. (2011), who used a set of heuristic rules to infer polarity shifts of discourse units based on their nuclearity status and outgoing relation links; Zirn et al. (2011), who used a lexicon-based sentiment system to predict the polarity scores of elementary discourse units and then enforced consistency of these assignments over the RST tree with the help of Markov logic constraints; and, finally, Wang and Wu (2013), who determined the semantic orientation of a document by taking a linear combination of the polarity scores of its EDUs and multiplying these scores with automatically learned coefficients.

Among the most recent advances in RST-aware sentiment research, we should especially emphasize the work of Bhatia et al. (2015), who proposed two different DASA systems:

- discourse-depth reweighting (DDR)
- and rhetorical recursive neural network (R2N2).

In the former approach, the authors estimated the relevance $\lambda_i$ of each elementary discourse unit $i$ as:

$$\lambda_i = \max\left(0.5, 1 - d_i/6\right),$$

where $d_i$ stands for the depth of the $i$-th EDU in the document's discourse tree. Afterwards, they computed the sentiment score $\sigma_i$ of that unit by taking the dot product of its binary feature vector $\mathbf{w}_i$ (token unigrams) with polarity scores $\boldsymbol{\theta}$ of these unigrams:

$$\sigma_i = \boldsymbol{\theta}^\top \mathbf{w}_i;$$

and then calculated the overall semantic orientation of the document $\Psi$ as the sum of sentiment scores for all units, multiplying these scores by their respective discourse-depth factors $\lambda$:

$$\Psi = \sum_i \lambda_i \boldsymbol{\theta}^T \mathbf{w}_i = \boldsymbol{\theta}^T \sum_i \lambda_i \mathbf{w}_i,$$

In the R2N2 system, the authors largely adopted the RNN method of Socher et al. (2013), recursively computing the polarity scores of discourse units as:

$$\psi_i = tanh\left(K_n^{(r_i)} \psi_{n(i)} + K_s^{(r_i)} \psi_{s(i)}\right),$$

where $K_n^{(r_i)}$ and $K_s^{(r_i)}$ stand for the nucleus and satellite coefficients associated with the rhetorical relation $r_i$, and $\psi_{n(i)}$ and $\psi_{s(i)}$ represent the sentiment scores of nucleus and satellite of the $i$-th vertex. This approach achieved 84.1% two-class accuracy on the movie review corpus of Pang and Lee (2004) and reached 85.6% on the dataset of Socher et al. (2013).

For the sake of completeness, we should note that there also exist discourse-aware sentiment approaches which build upon PDTB and SDRT. For example, Trivedi and Eisenstein (2013) proposed a method based on latent structural SVM (Yu and Joachims, 2009), where they represented each sentence as a vector of features produced by a feature function $\mathbf{f}(y, \mathbf{x}_i, h_i)$, where $y \in \{-1, +1\}$ denotes the potential polarity of the whole document, $h_i \in \{0, 1\}$ stands for the assumed subjectivity class of sentence $i$, and $\mathbf{x}_i$ represents the surface form of that sentence; and then tried to infer the most likely semantic orientation of the document $\hat{y}$ over all possible assignments $\mathbf{h}$, *i.e.*,:

$$\hat{y} = \underset{y}{\operatorname{argmax}} \left( \max_{\mathbf{h}} \mathbf{w}^\top \mathbf{f}(y, \mathbf{x}, \mathbf{h}) \right).$$

To ensure that these assignments were still coherent, the authors additionally extended their feature space with special *transitional* attributes which indicated whether two adjacent sentences were likely to share the same subjectivity given the discourse connective between them. With the help of these features, Trivedi and Eisenstein could improve the accuracy of the connector-unaware model on the movie review corpus of Maas et al. (2011) from 88.21 to 91.36%.

The first step towards an SDRT-based sentiment approach was made by Asher et al. (2008), who presented an annotation scheme and a pilot corpus of English and French texts, which were analyzed according to the SDRT theory and enriched with additional sentiment information. Specifically, the authors asked the annotators to ascribe one of four opinion categories (reporting, judgment, advice, or sentiment) along with their subclasses (e.g., inform, assert, blame, recommend) to each discourse unit which had at least one opinionated word from a sentiment lexicon. Afterwards, they showed that with a simple set of rules, one could easily propagate opinions through SDRT graphs, increasing the strengths or reversing the

polarity of the sentiments, depending on the type of the discourse relation that was linking two segments.

In general, however, PDTB- and SDRT-based sentiment systems are much less common than RST-inspired solutions. Because of this fact and due to the reasons described in Section 7.1, we will primarily concentrate on this last group of methods. In particular, for the sake of comparison, we replicated the linear combination approach of Wang and Wu (2013) and also reimplemented the DDR and R2N2 systems of Bhatia et al. (2015). Furthermore, to see how these techniques would perform in comparison with much simpler baselines, we additionally created two methods which predicted the polarity of a message by only considering its last or topmost nucleus EDU (henceforth LAST and ROOT), and also estimated the results of the original LBA classifier without any discourse-related modifications (henceforth NO-DISCOURSE). Apart from these baselines and existing methods, we propose a few novel DASA solutions, which will be briefly described below.

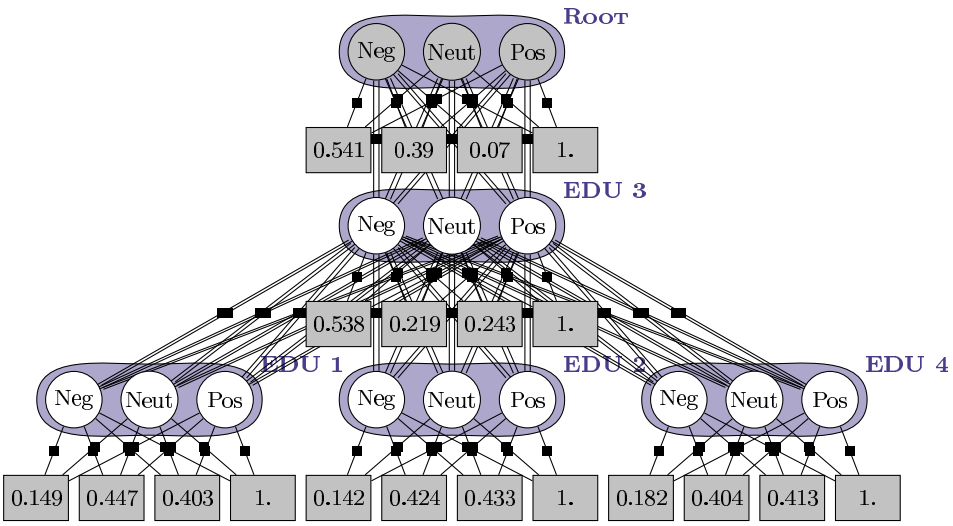
**Latent CRF.** In the first of these solutions, called *Latent Conditional Random Fields* or *LCRF*, we consider the problem of message-level sentiment analysis as an inference task over undirected graphical model, where the nodes of the model represent polarity probabilities of elementary discourse units and the structure of the graph reflects the RST dependency tree of the message.[4] In particular, we define CRF graph $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ as a set of vertices $\mathcal{V} := \mathcal{Y} \cup \mathcal{X}$, in which $\mathcal{Y} := \{y_{(i,j)} \mid i \in \{\text{ROOT}, 1, 2, \ldots, T\}, j \in \{\text{NEGATIVE}, \text{NEUTRAL}, \text{POSITIVE}\}\}$ represents (partially observed) random variables (with $T$ standing for the number of EDUs in the tweet), and $\mathcal{X} := \{x_{(i,j)} \mid i \in \{\text{ROOT}, 1, 2, \ldots, T\}, j \in [0, \ldots, 3]\}$ denotes the respective features of these nodes (three polarity scores returned by the LBA classifier plus an additional offset feature whose value is always `1` irrespectively of the input). Since the ROOT vertex, however, does not have a corresponding discourse segment in the RST tree, we use the polarity scores predicted by the LBA classifier for the whole message as features of this node.

Graph edges $\mathcal{E}$ connect random variables to their corresponding features and also link every pair of vertices $(v_{(k,\cdot)}, v_{(i,\cdot)})$ if node $k$ appears as the parent of node $i$ in the RST dependencies.[5] You can see an actual example of such automatically induced CRF tree in Figure 7.6.

Now, before we describe the training of our model, let us briefly recall that in the standard CRF optimization we typically try to find optimal parameters $\boldsymbol{\theta}^*$ which maximize the log-

[4]Drawing on the work of Bhatia et al. (2015), we obtain this representation using the DEP-DT algorithm of Hirao et al. (2013) with a minor modification that we do not follow any satellite branches while computing the heads of abstract RST nodes in Step 1 of this procedure (see Hirao et al., 2013, pp. 1516–1517).

[5]In fact, we use two edges to connect each child to its parent: one for the CONTRASTIVE relation and another one for the NON-CONTRASTIVE link.




[Gucke Lost]$_1$ [und esse Obst .]$_2$ [Fühlt sich fast an ,]$_3$ [als wäre das mein Leben .]$_4$

[*Watching Lost*]$_1$ [*and eating fruits* .]$_2$ [*Almost feels*]$_3$ [*as if it were my life* .]$_4$

*Figure 7.6: Example of an automatically constructed RST-based latent-CRF tree (random variables are shown as circles, fixed input parameters appear as rectangles, and observed values are displayed in gray)*

likelihood of all label sequences $\mathbf{y}^{(i)}$ on the training set $\mathcal{D} := \left\{\left(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\right)\right\}_{i=1}^{N}$, *i.e.*:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \ell(\boldsymbol{\theta}) = \sum_{i=1}^{N} \log\left(p\left(\mathbf{y}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}\right)\right),$$

where the conditional likelihood is normally estimated as:

$$p\left(\mathbf{y}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}\right) = \frac{\exp\left(\sum_{t=1}^{T_i}\sum_k \boldsymbol{\theta}_k \mathbf{f}_k\left(\mathbf{x}_t^{(i)}, \mathbf{y}_{t-t}^{(i)}, \mathbf{y}_t^{(i)}\right)\right)}{Z}.$$

Adapting this equation to our RST-based CRF structures, we obtain:

$$p\left(\mathbf{y}^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}\right) = \frac{\exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t^{(i)}, \mathbf{y}_c^{(i)}\right)\right]\right)}{Z}, \quad (7.1)$$

where $ch(t)$ denotes the children of node $t$, $v$ stands for the indices of node features, and $e$ represents the indices of edge attributes.

A crucial problem with this formulation though is that in our task, only a small subset of labels from $\mathbf{y}^{(i)}$ (namely those of the root node) are actually observed at training time, whereas the rest of the tags (those which pertain to EDUs) are unknown. We will denote these observed and hidden subsets as $\mathbf{y}_o^{(i)}$ and $\mathbf{y}_h^{(i)}$ respectively. Using this notation, we can redefine the training objective of our model as finding such parameters $\boldsymbol{\theta}^*$ which maximize the log-likelihood of *observed* labels, *i.e.*:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \log\left(p\left(\mathbf{y}_o^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}\right)\right).$$

With this formulation, however, it is still unclear what we should do with hidden tags $\mathbf{y}_h^{(i)}$, because the values of their features remain undefined.

One possible way to approach the problem of unobserved states in the input is to assume that any label sequence $\mathbf{y}_h^{(i)}$ might be true, and then try to maximize the parameters along the path which leads to the maximum probability of the correct observed tag, *i.e.*:

$$\begin{aligned} \mathbf{y}^{(i)} &:= [\mathbf{y}_o^{(i)}, \mathbf{y}_h^{*(i)}], \text{ where} \\ \mathbf{y}_h^{*(i)} &= \underset{\mathbf{y}_h^{(i)}}{\operatorname{argmax}}\, p\left(\mathbf{y}_o^{(i)} | \mathbf{x}^{(i)}\right), \end{aligned} \tag{7.2}$$

and which we can easily find using the standard Viterbi decoding.

Unfortunately, if we simply consider label sequence $\mathbf{y}^{(i)}$ from Equation 7.2 as the ground truth and penalize all labels which disagree with this sequence, we might overly commit ourselves to the model's guess of unknown tags and unduly discriminate against other possible hidden label assignments. To mitigate this effect, we can instead penalize only one other sequence, namely, that which maximizes the probability of an incorrect label at the observed state:

$$\begin{aligned} \mathbf{y}^{'(i)} &:= \underset{\mathbf{y}_o^{'(i)} \neq \mathbf{y}_o^{(i)}}{\operatorname{argmax}}\, p\left([\mathbf{y}_o^{'(i)}, \mathbf{y}_h^{*(i)}] | \mathbf{x}^{(i)}\right), \text{ where} \\ \mathbf{y}_h^{*(i)} &= \underset{\mathbf{y}_h^{(i)}}{\operatorname{argmax}}\, p\left(\mathbf{y}_o^{'(i)} | \mathbf{x}^{(i)}\right). \end{aligned}$$

Correspondingly, we reformulate our objective and instead of maximizing the log-likelihood of the training set will now maximize the difference between the log-probabilities of the correct and most likely wrong assignments:

$$\begin{aligned} \boldsymbol{\theta}^* &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \log\left(p\left(\mathbf{y}^{(i)}\right)\right) - \log\left(p\left(\mathbf{y}^{'(i)}\right)\right) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \log\left(\exp\left(\boldsymbol{\theta}^\top \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\right)\right) - \log\left(\exp\left(\boldsymbol{\theta}^\top \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{'(i)})\right)\right) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{i=1}^{N} \boldsymbol{\theta}^\top \left(\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{'(i)})\right), \end{aligned} \tag{7.3}$$

where $\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ and $\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{'(i)})$ mean all features associated with label sequences $\mathbf{y}^{(i)}$ and $\mathbf{y}^{'(i)}$ respectively.

The only thing that we now need to do to the above objective is to introduce a regularization term $\frac{1}{2}\|\boldsymbol{\theta}\|^2$ in order to prevent its divergence to infinity in cases when $\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ and $\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{'(i)})$ are perfectly separable. This brings us to the final formulation:

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{\theta}\|^2 - \sum_{i=1}^{N} \boldsymbol{\theta}^\top \left(\mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{'(i)})\right) \tag{7.4}$$

At this point, we can notice that the resulting function is identical to the unconstrained minimization problem of structural SVM (Taskar et al., 2003), and we indeed can piggyback on one of the many efficient SVM-optimization techniques to learn the parameters of our model. In particular, we use the block-coordinate Frank-Wolfe algorithm (Lacoste-Julien et al., 2013), running it for 1,000 epochs or until convergence, whichever of these events occurs first.

**Latent-Marginalized CRF.** Another way to tackle unobserved labels is to estimate the probability of observed tags by marginalizing (summing) out hidden variables from the joint distribution, *i.e.*:

$$p\left(\mathbf{Y}_o{=}\mathbf{y}_o\right) = \sum_{\mathbf{y}_h} p\left(\mathbf{Y}_o{=}\mathbf{y}_o, \mathbf{Y}_h{=}\mathbf{y}_h\right).$$

Applying this formula to Equation 7.1, we get:

$$\begin{aligned} p\left(\mathbf{y}_o^{(i)}|\mathbf{x}^{(i)};\boldsymbol{\theta}\right) &= \sum_{\mathbf{y}_h^{(i)}} p\left([\mathbf{y}_o^{(i)}, \mathbf{y}_h^{(i)}]|\mathbf{x}^{(i)};\boldsymbol{\theta}\right) \\ &= \frac{\sum_{\mathbf{y}_h^{(i)}} \exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t^{(i)}, \mathbf{y}_c^{(i)}\right)\right]\right)}{Z}, \end{aligned}$$

where $\mathbf{y}^{(i)}$ in the numerator is defined as before: $\mathbf{y}^{(i)} := [\mathbf{y}_o^{(i)}, \mathbf{y}_h^{(i)}]$.

This time again, we would like to maximize the probability of the correct assignment, setting it apart from its closest competitor by some margin. Unfortunately, due to the summation over all $\mathbf{y}_h^{(i)}$, we cannot avail ourselves of the log-exp cancellation trick which we used previously in Equation 7.3. Instead of this, we replace the difference of the log-likelihoods by the ratio of marginal probabilities:

$$\begin{aligned} \boldsymbol{\theta}^* &= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{i=1}^{N} \frac{p(\mathbf{y}^{(i)})}{p(\mathbf{y}'^{(i)})} \\ &= \operatorname*{argmax}_{\boldsymbol{\theta}} \sum_{i=1}^{N} \frac{\sum_{\mathbf{y}_h^{(i)}} \exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t^{(i)}, \mathbf{y}_c^{(i)}\right)\right]\right)}{\sum_{\mathbf{y}_h^{(i)}} \exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t'^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t'^{(i)}, \mathbf{y}_c'^{(i)}\right)\right]\right)} \end{aligned} \tag{7.5}$$

To simplify this expression, we can introduce the following abbreviations:

$$\begin{aligned} a &:= \exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t^{(i)}, \mathbf{y}_c^{(i)}\right)\right]\right), \\ b &:= \exp\left(\sum_{t=0}^{T_i}\left[\sum_v \boldsymbol{\theta}_v \mathbf{f}_v\left(\mathbf{x}_t^{(i)}, \mathbf{y}_t'^{(i)}\right) + \sum_{c\in ch(t)}\sum_e \boldsymbol{\theta}_e \mathbf{f}_e\left(\mathbf{y}_t'^{(i)}, \mathbf{y}_c'^{(i)}\right)\right]\right). \end{aligned}$$

Now, we estimate the derivatives of functions $a$ and $b$ w.r.t. a single parameter $\boldsymbol{\theta}_v$ as:

$$\frac{\partial a}{\partial \boldsymbol{\theta}_v} = a \sum_{t=0}^{T_i} \mathbf{f}_v \left( \mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)} \right) \propto \mathbb{E}_{\mathbf{y}^{(i)}} \mathbf{f}_v,$$
$$\frac{\partial b}{\partial \boldsymbol{\theta}_v} = b \sum_{t=0}^{T_i} \mathbf{f}_v \left( \mathbf{x}_t^{(i)}, \mathbf{y}_t^{(i)} \right) \propto \mathbb{E}_{\mathbf{y}'^{(i)}} \mathbf{f}_v;$$

and analogously obtain:

$$\frac{\partial a}{\partial \boldsymbol{\theta}_e} = a \sum_{t=0}^{T_i} \sum_{c \in ch(t)} \mathbf{f}_e \left( \mathbf{y}_t^{(i)}, \mathbf{y}_c^{(i)} \right) \propto \mathbb{E}_{\mathbf{y}^{(i)}} \mathbf{f}_e,$$
$$\frac{\partial b}{\partial \boldsymbol{\theta}_e} = b \sum_{t=0}^{T_i} \sum_{c \in ch(t)} \mathbf{f}_e \left( \mathbf{y}_t^{\prime(i)}, \mathbf{y}_c^{\prime(i)} \right) \propto \mathbb{E}_{\mathbf{y}'^{(i)}} \mathbf{f}_e.$$

With the help of these expressions, we can easily compute the gradient of the objective function w.r.t. $\boldsymbol{\theta}$ by observing that:

$$\nabla_{\boldsymbol{\theta}} = \sum_{i=1}^{N} \frac{\sum_{\mathbf{y}_h^{(i)}} \nabla_{\boldsymbol{\theta}} a \sum_{\mathbf{y}_h^{(i)}} b - \sum_{\mathbf{y}_h^{(i)}} a \sum_{\mathbf{y}_h^{(i)}} \nabla_{\boldsymbol{\theta}} b}{\left( \sum_{\mathbf{y}_h^{(i)}} b \right)^2}. \tag{7.6}$$

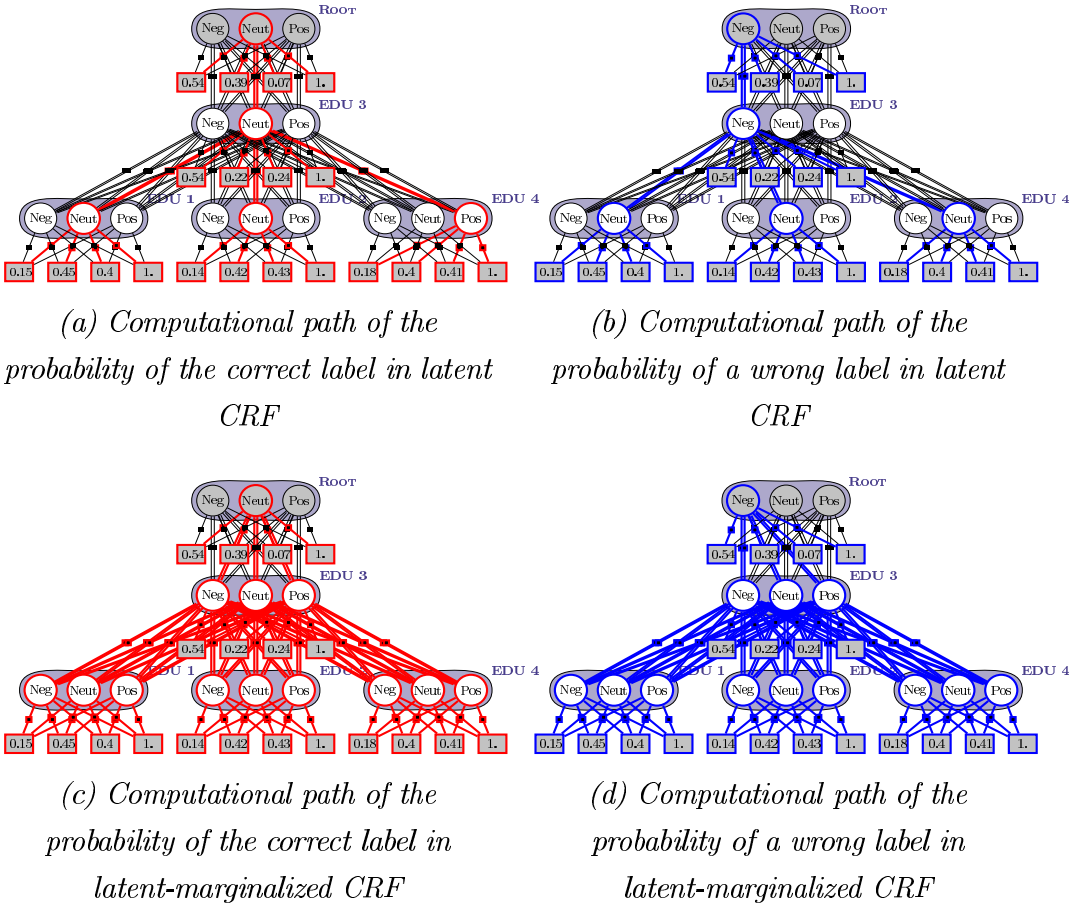
We again use the block-coordinate Frank-Wolfe algorithm to optimize the parameters of our model, but instead of pushing these parameters in the direction $\boldsymbol{\psi} := \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}) - \mathbf{f}(\mathbf{x}^{(i)}, \mathbf{y}'^{(i)})$ (which is the derivative of latent CRF, see Algorithm 2 in [Lacoste-Julien et al., 2013]), we now maximize them along the gradient from Equation 7.6.

It is probably easier to realize the difference between the two CRF methods (latent and latent-marginalized CRFs) more vividly by looking at Figure 7.7, in which we have highlighted the paths that are used to compute the probabilities of correct and wrong labels in both systems. As we can see from this picture, LCRF only considers one label sequence that leads to the maximum probability of the correct tag (NEUT) at the single observed ROOT node and then compares this sequence with the path that maximizes the probability of an incorrect tag (in this case NEG) at the same node. In contrast to this, LMCRF considers all possible label configurations of elementary discourse units and uses this total cumulative mass to estimate the probability of both (correct and wrong) observed tags.

**Recursive Dirichlet Process.** Finally, the last method that we present in this chapter—*Recursive Dirichlet Process* or *RDP*—goes a further step in the probabilistic direction by assuming that not only the probabilities of discourse units but also the parameters via which these probabilities are computed represent random variables.

In particular, we associate a variable $\mathbf{z}_j \in \mathbb{R}^3_+$, s.t. $\|\mathbf{z}\|_1 = 1$, with every RST node $i$ (which in this case can be either an elementary discourse segment or an abstract span).[6] This

[6] In contrast to the previous CRF approaches, this time, we depart from the dependency tree represen-




*(a) Computational path of the probability of the correct label in latent CRF*

*(b) Computational path of the probability of a wrong label in latent CRF*

*(c) Computational path of the probability of the correct label in latent-marginalized CRF*

*(d) Computational path of the probability of a wrong label in latent-marginalized CRF*

*Figure 7.7: Confronted computational paths in latent and latent-marginalized conditional random fields*

variable specifies the multivariate probability of the three polarities (NEGATIVE, NEUTRAL, and POSITIVE) for the $j$-th node. Since every element of $\mathbf{z}_j$ has to be non-negative and their total sum must add up to one, it is natural to assume that the value of this variable is drawn from a Dirichlet distribution:

$$\mathbf{z}_j \sim Dir(\boldsymbol{\alpha}).$$

The only parameter accepted by this distribution, which simultaneously controls both the mean and the variance of its outcomes, is vector $\boldsymbol{\alpha}$. Consequently, our primary goal in this method is to find a way how to compute this parameter automatically for each node.

An obvious starting point for this computation is the polarity scores predicted by the base classifier for every elementary discourse unit, which we will henceforth denote as $\boldsymbol{z}_{j_0} \in \mathbb{R}^3_+$. Since these scores, however, are only available for elementary segments, we initialize the corresponding variables of the abstract spans to zeroes with the only exception being the root node, to which we again assign the scores returned by the LBA classifier for the whole

tation and adopt the discourse tree structure proposed by Bhatia et al. (2015) for their R2N2 method. In this structure, we keep all abstract nodes from the original RST tree, but relink all satellites to the abstract parents of their nuclei.

message.

To compute the posterior distribution of the root ($\mathbf{z}_{\text{ROOT}}$), we sort all nodes of the RST tree in reverse topological order and estimate the polarities of the spans from the bottom up by joining the **z**-scores of their children. However, before we do this joining, we multiply the **z**-vector of each child $k$ with a special matrix $M_r$, where $r \in \{\{\text{NUCLEUS}, \text{SATELLITE}\} \times \{\text{CONTRASTIVE}, \text{NON-CONTRASTIVE}\}\}$ is the discourse relation holding between that child and its parent, and project the result of this multiplication back to the probability simplex using the sparsemax operation (Martins and Astudillo, 2016):

$$\mathbf{z}_k^* = sparsemax\left(M_r \mathbf{z}_k^\top\right). \tag{7.7}$$

The main goal of matrix $M_r$ is to reflect contextual polarity changes that might be conveyed by discourse relations; for example, a contrastive link might stronger affect the polarity of the parent than the non-contrastive one (compare, for instance, the contrastive *Many people support Trump, but he behaves like an alpha male* with the non-contrastive *Many people support Trump, because he behaves like an alpha male*). Because this parameter also represents a random variable, we sample it from a multivariate normal distribution:

$$M_r \sim \mathcal{N}_{3\times 3}(\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r),$$

setting the mean of this distribution to:[7]

$$\boldsymbol{\mu}_r := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.3 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

and initializing its covariance matrix to all ones:

$$\boldsymbol{\Sigma}_r := \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

With this choice of parameters, we hope to dampen the effect of neutral EDUs[8] in order to prevent situations when multiple objective segments vanquish the meaning of a single polar discourse unit.

Afterwards, when seeing the $k$-th child of the $j$-th node in the RST tree, we compute the $\boldsymbol{\alpha}$ parameter of this node using the following formula:

$$\boldsymbol{\alpha}_{j_k} = \boldsymbol{\beta} \odot \mathbf{z}_k^* + (\mathbf{1} - \boldsymbol{\beta}) \odot \mathbf{z}_{j_{k-1}}, \tag{7.8}$$

[7] Before we do the actual sampling, we unroll this parameter to a vector and then reshape the sampled value back to a $3 \times 3$ matrix.

[8] As you can see from Equation 7.7, the middle row of the $M_r$ matrix is responsible for propagating the neutral score of the $j$-the node, and by setting this row to $[0, 0.3, 0]$ we effectively reduce the neutral polarity by two thirds.

where $\boldsymbol{\beta} \in \mathbb{R}^3$ is another multivariate random variable sampled from the Beta distribution $B(5., 5.)$, which controls the amount of information we want to pass from child to its parent; $\mathbf{z}_{j_{k-1}}$ is the value of the $\mathbf{z}$-vector for the $j$-th node after seeing its previous child; and $\odot$ means elementwise multiplication.

The only thing that we now need to do to the above $\boldsymbol{\alpha}_{j_k}$ term before we draw the actual probability $\boldsymbol{z}_{j_k}$ is to scale this vector by a certain amount in order to reduce the variance of the resulting Dirichlet distribution.[9] In particular, we compute this scaling factor as follows:

$$scale = \frac{\xi \times \left(0.1 + \cos\left(\mathbf{z}_k^*, \mathbf{z}_{j_{k-1}}\right)\right)}{H\left(\boldsymbol{\alpha}_{j_k}\right)};$$

where $\xi$ is a model parameter sampled from a $\chi^2$-distribution: $\xi \sim \chi^2(34)$; 0.1 is a constant used to prevent zero scales in the cases when $\cos\left(\mathbf{z}_k^*, \mathbf{z}_{j_{k-1}}\right)$ is zero; and $H\left(\boldsymbol{\alpha}_{j_k}\right)$ stands for the entropy of the $\boldsymbol{\alpha}_{j_k}$ vector. Although this expression looks somewhat complicated, the intuition behind it is very simple: The $\xi$ term encodes our prior belief in the correctness of model's predictions (the higher its value, the more we trust the model); the cosine measures the similarity between the probabilities of parent and child (the more similar these probabilities, the greater will be the scale); and, finally, the entropy in the denominator tells us how uniform the vector $\boldsymbol{\alpha}_{j_k}$ is (the more equal its scores, the less confident we will be in the final outcome).

With the *scale* and $\boldsymbol{\alpha}_{j_k}$ terms at hand, we are all set to compute the updated probability of polar classes for the $j$-th node after considering its $k$-th child:

$$\mathbf{z}_{j_k} \sim Dir(scale \times \boldsymbol{\alpha}_{j_k}).$$

You can see some examples of this computation in Figure 7.8, where we have plotted different configurations of parent and child probabilities ($\mathbf{z}_{j_{k-1}}$ and $\mathbf{z}_k$, shown to the right of each picture) and the resulting Dirichlet distributions (represented as simplices). For instance, in the top-left figure, we show a situation where the parent has a very strong probability of the negative class ($[1, 0, 0]$), but the probability of the child is absolutely uniform ($[0.33, 0.33, 0.33]$); in this case, the model keeps to the negative polarity, heaping almost all probability mass in this corner. At the same, to account for the uncertainty about the child, RDP slightly moves the crest of the probability hill (*i.e.*, its mean) towards the positive class and makes the slopes of this hill lower along all three axes (*i.e.*, increases its variance). On the other hand, if parent and child have completely opposite semantic orientations (say POSITIVE and NEGATIVE), which the base classifier is perfectly sure about, as shown in Subfigure b, RDP uniformly distributes the whole probability just along the narrow POSITIVE–NEGATIVE edge. Another situation is depicted in the middle row, where

[9] Because if we keep the $\boldsymbol{\alpha}_{j_k}$ vector from Equation 7.8 unchanged, most of its values will be in the range $[0, \ldots, 1]$ which will lead to an extremely high variance of the Dirichlet distribution.

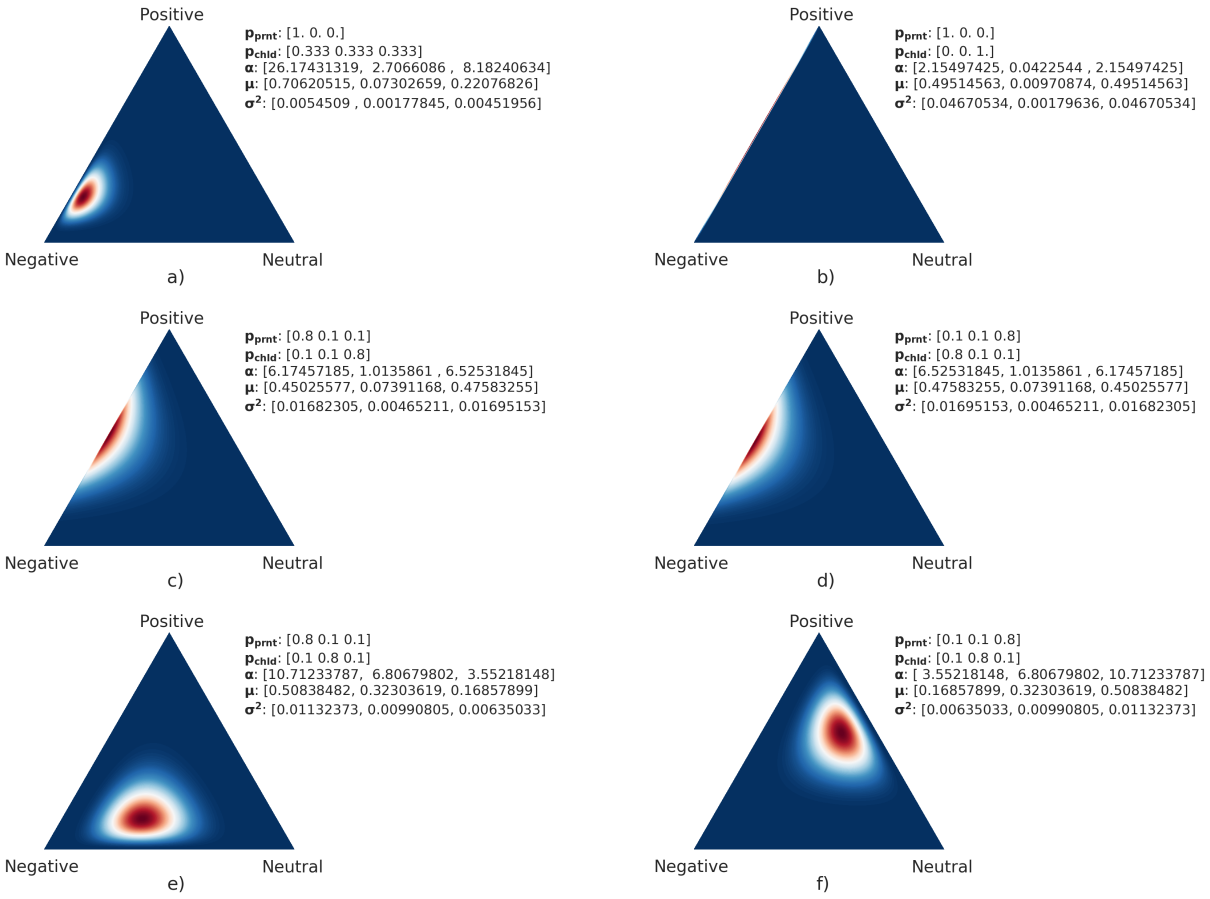



*Figure 7.8: Probability distributions of polar classes computed by the Recursive Dirichlet Process (higher probability regions are highlighted in red; $\mathbf{p}_{prnt}$ means the probability of the parent node [the values in the vector represent the scores for the negative, neutral, and positive polarities respectively]; $\mathbf{p}_{chld}$ denotes the probability of the child; and $\boldsymbol{\alpha}$, $\boldsymbol{\mu}$, and $\boldsymbol{\sigma}^2$ represent the parameters of the resulting joint distribution shown in the simplices)*

parent and child again have opposite polarities, but the base predictor is less sure about its decisions and also admits a small chance that either of these nodes is neutral. In this case, RDP still spreads most probability along the main polar edge, but places the mean of this distribution right in-between the two polar corners and also screeds some part of that mass towards the center of the simplex. Finally, in the last row, we can see our intended discrimination of the neutral orientation: This time, the parent node is strictly polar (negative on the left and positive on the right), whereas its child is neutral. In contrast to the previous two examples, the mean of the resulting distribution is located closer to the polar corner and not in-between the two juxtaposed classes as before.

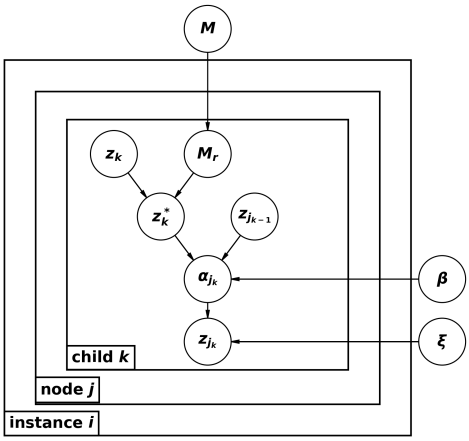



*Figure 7.9: Plate diagram of the Recursive Dirichlet Process (without the final categorical draw)*

Returning back to our model, after processing all $K$ children of the $j$-th node, we regard the last outcome $\mathbf{z}_{j_K}$ as the final polarity distribution of that node and use this value to estimate the probabilities of the remaining ancestors in the RST tree. Finally, after finishing processing all descendants of the root, we use the resulting $\mathbf{z}_{\text{ROOT}_K}$ vector as a parameter of a categorical distribution from which we draw the final prediction label $y$:

$$y \sim Cat(\mathbf{z}_{\text{ROOT}}).$$

Using this manually defined model with its hand-tuned fixed parameters, we can estimate our prior belief in the joint probability of hidden and observed variables $p(y, \mathbf{z})$. As it turns out, knowing this belief is enough to derive another probability $q(\mathbf{z})$, which best approximates the distribution of only the latent nodes. In particular, we define the structure of $q(\mathbf{z})$ to be the same as in $p(y, \mathbf{z})$, but deprive it of the last step (drawing of the observed label) and optimize the parameters $\boldsymbol{\theta}$ of this model ($\boldsymbol{\mu}_r$, $\boldsymbol{\Sigma}_r$, and the parameters of the Beta and $\chi^2$ distributions) by maximizing the evidence lower bound between $p$ and $q$, using stochastic gradient descent(see Ranganath et al., 2014):

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z})}\left[\log\left(p(y, \mathbf{z})\right) - \log\left(q(\mathbf{z})\right)\right].$$

We perform this optimization for 100 epochs, picking the parameters which yield the best macro-averaged $F_1$-score on the set-aside development data.

The results of our proposed and baseline methods are shown in Table 7.1.

As we can see from the table, our approaches perform fairly well in comparison with other systems, outperforming them in terms of macro- and macro-averaged $F_1$ on both datasets. Especially the latent-marginalized CRF shows fairly strong scores, yielding the best $F_1$-results for the positive and neutral classes on the PotTS and SB10k data, which in turn

| Method | Positive | | | Negative | | | Neutral | | | Macro | Micro |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | $F_1$ | Precision | Recall | $F_1$ | Precision | Recall | $F_1$ | $F_1$ | $F_1$ |
| PotTS | | | | | | | | | | | |
| LCRF | 0.76 | 0.79 | **0.77** | **0.61** | 0.53 | 0.56 | 0.7 | 0.71 | 0.71 | 0.67 | 0.709 |
| LMCRF | **0.77** | 0.77 | **0.77** | **0.61** | 0.54 | 0.57 | 0.69 | **0.74** | **0.72** | 0.671 | **0.712** |
| RDP | 0.73 | 0.82 | **0.77** | **0.61** | 0.56 | **0.58** | **0.73** | 0.65 | 0.69 | **0.678** | 0.706 |
| DDR | 0.73 | 0.77 | 0.75 | 0.54 | **0.59** | 0.56 | 0.69 | 0.61 | 0.65 | 0.655 | 0.674 |
| R2N2 | 0.74 | 0.78 | 0.76 | 0.59 | 0.53 | 0.56 | 0.68 | 0.68 | 0.68 | 0.657 | 0.692 |
| WNG | 0.58 | 0.79 | 0.67 | **0.61** | 0.21 | 0.31 | 0.61 | 0.57 | 0.59 | 0.487 | 0.59 |
| Last | 0.52 | **0.83** | 0.64 | 0.57 | 0.17 | 0.26 | 0.61 | 0.43 | 0.5 | 0.453 | 0.549 |
| Root | 0.56 | 0.73 | 0.64 | 0.58 | 0.22 | 0.32 | 0.55 | 0.54 | 0.54 | 0.481 | 0.56 |
| No-Discourse | 0.73 | 0.82 | **0.77** | **0.61** | 0.56 | **0.58** | 0.72 | 0.66 | 0.69 | 0.677 | 0.706 |
| SB10k | | | | | | | | | | | |
| LCRF | **0.64** | **0.69** | 0.66 | 0.45 | **0.45** | 0.45 | **0.82** | 0.79 | **0.8** | 0.557 | 0.713 |
| LMCRF | **0.64** | **0.69** | **0.67** | 0.45 | **0.45** | 0.45 | **0.82** | 0.79 | **0.8** | **0.56** | **0.715** |
| RDP | **0.64** | **0.69** | 0.66 | 0.45 | **0.45** | 0.45 | 0.82 | 0.79 | **0.8** | 0.557 | 0.713 |
| DDR | 0.59 | 0.63 | 0.61 | **0.48** | 0.44 | **0.46** | 0.77 | 0.76 | 0.77 | 0.534 | 0.681 |
| R2N2 | **0.64** | **0.69** | 0.66 | 0.46 | **0.45** | 0.45 | 0.81 | 0.79 | **0.8** | 0.559 | 0.713 |
| WNG | 0.61 | 0.63 | 0.62 | 0.46 | 0.29 | 0.36 | 0.76 | **0.82** | 0.79 | 0.488 | 0.693 |
| Last | 0.56 | 0.55 | 0.56 | 0.46 | 0.29 | 0.36 | 0.73 | 0.8 | 0.76 | 0.459 | 0.661 |
| Root | 0.51 | 0.55 | 0.53 | 0.4 | 0.3 | 0.35 | 0.74 | 0.76 | 0.75 | 0.438 | 0.64 |
| No-Discourse | **0.64** | **0.69** | 0.66 | 0.45 | **0.45** | 0.45 | **0.82** | 0.79 | **0.8** | 0.557 | 0.713 |

*Table 7.1: Results of discourse-aware sentiment analysis methods*
*LCRF – latent conditional random fields, LMCRF – latent-marginalized conditional random fields, RDP – recursive Dirichlet process, DDR – discourse-depth reweighting (Bhatia et al., 2015), R2N2 – rhetorical recursive neural network (Bhatia et al., 2015), WNG – Wang and Wu (2013),* Last *– polarity determined by the last EDU,* Root *– polarity determined by the root EDU(s),* No-Discourse *– discourse-unaware classifier*

leads to the highest overall micro-averaged $F_1$-measure on these corpora. This solution is closely followed by the Recursive Dirichlet Process, whose $F_1$ for the positive class on the PotTS test set is identical to that attained by LMCRF and the $F$-score for the negative class is even one percent higher, which allows it to reach the best macro-average on this test set.

As it turns out, the strongest competitors to our systems are the No-Discourse approach and the R2N2 method by Bhatia et al. (2015). The former solution outperforms the latter on the PotTS corpus on both metrics (macro- and micro-$F_1$), but falls against it with respect to the macro-$F_1$ on the SB10k set. The DDR and WNG methods get sixth and seventh places respectively, followed by the simplest solutions—Last and Root. Interestingly enough, the Last approach beats the Root method on the SB10k data, but shows worse scores on the PotTS corpus, which is mostly due to the lower recall of the negative class.

### 7.3.1 Error Analysis

Although our methods performed quite competitive, we decided to still have a closer look at their errors in order to understand their remaining potential weaknesses better.

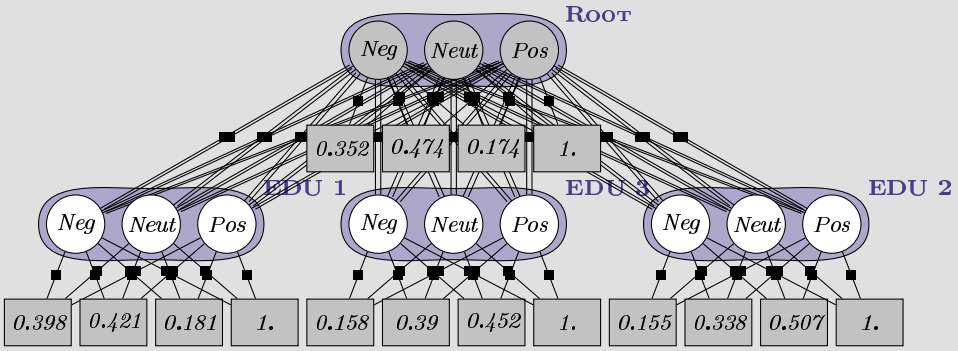
**Example 7.3.2 (Error Made by the LCRF System)**

**Tweet:** [Boah , also doch wieder ein Mann , oder ?]$_1$ [ODER ?]$_2$ [papst]$_3$

[*Damn, a man again, isn't it ?*]$_1$ [*ISN'T ?*]$_2$ [*pope*]$_3$

**Gold Label:** negative

**Predicted Label:** neutral*




The first such error shown in Example 7.3.2 was made by the latent CRF system, which erroneously considered an evidently negative tweet as neutral. But as we can see from the picture of the automatic RST tree in this example, we can hardly expect the right decision in this case anyway, because neither EDUs nor the root node of this message had been correctly classified as negative by the LBA classifier. Nevertheless, even in this apparently hopeless situation, messages propagated from the leaves to the root during the max-product inference still tell the latter node that the predicted class better be negative. (We inspected the belief propagation messages passed in the forward direction and found that the total score for the negative class amounts to 0.597, whereas the belief in the positive class [its closest rival] only runs up to 0.462.) Unfortunately, these messages cannot outweigh the high score of the neutral class that results from the node features (the state score for this polarity is equal to 0.524, whereas the negative class only obtains a score of -0.118).[10]

As it turns out, high neutral node scores of the root are also the main reason for the misclassification in Example 7.3.3, where the LMCRF system also confuses the negative polarity with the neutral class. This time, however, messages coming from the leaves suggest almost equal probabilities for both semantic orientations, so that feature scores of the root completely call the shots in the final decision.

[10] All scores for this example are given in the logarithm domain.

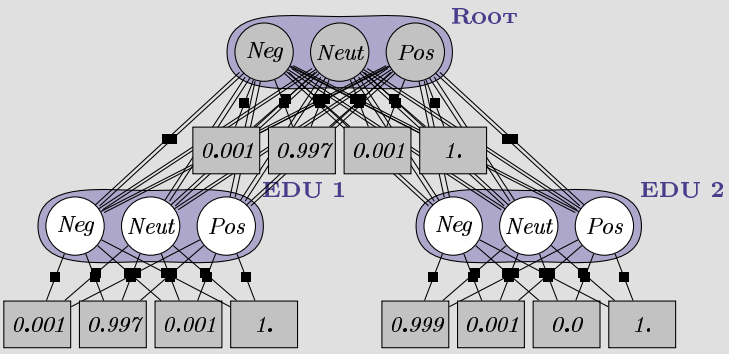
**Example 7.3.3 (Error Made by the LMCRF System)**

**Tweet:** ’ [Wissen ?]$_1$ [Igitt geh weg damit !]$_2$

[*Knowledge ?*]$_1$ [*Yuck , go away with it*]$_2$

**Gold Label:** **negative**

**Predicted Label:** **neutral***




Unfortunately, the recursive Dirichlet process cannot withstand the erroneous predictions of the base classifier either. For instance, in Example 7.3.4, LBA assigns the highest scores to the positive class in three out of four EDUs, even though each of these units by itself expresses a negative (sarcastic) attitude of the author. Alas, the only case where the base classifier correctly predicts the negative label (“Das is noch lange nicht ausdiskutiert !” [*It’s no way been talked out !*]) drowns at the very beginning of the score propagation. (As it turned out, the learned $\beta$ parameter, which controls the amount of information passed from child to its parent in Equation 7.8, is extremely low for the negative class, amounting to only 0.097, whereas for the positive and negative polarities it runs up to 0.212 and 0.279. Due to this low value, only one tenth of the negative score from the third EDU arrives at the parent when the model computes the polarity scores of the abstract span 2.)

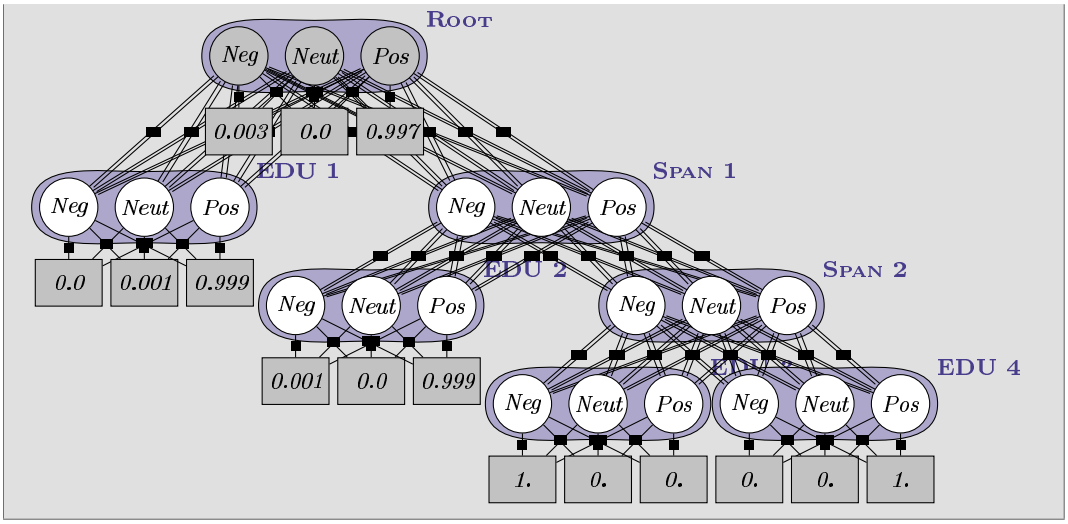
**Example 7.3.4 (Error Made by the RDP System)**

**Tweet:** [Prima , was sind das für Idioten im DFB ?]$_1$ [Das is eine Muppetsshow auf LSD !]$_2$ [Das is noch lange nicht ausdiskutiert !]$_3$ [Kiessling ist ein Depp !]$_4$

[*Great, who are these idiots in the DFB ?*]$_1$ [*It is a muppet show on LSD*]$_2$ [*It’s no way been talked out !*]$_3$ [*Kiessling is a goof !*]$_3$

**Gold Label:** **negative**

**Predicted Label:** **positive***




Another interesting error shown in Example 7.3.5 was made by the baseline ROOT system, which similarly to CRF-based approaches confused the negative class with the neutral polarity. This time, however, the misclassification is due to the discourse structure itself rather than wrong predictions of the underlying sentiment method. Because LBA correctly recognizes that the negative smiley at the end of tweet has a strictly negative semantic orientation, but the discourse-aware baseline does not see this EDU at all, but instead only considers the segment at the top of the tree, which merely expresses a factual hypothesis, free of any polar connotation.

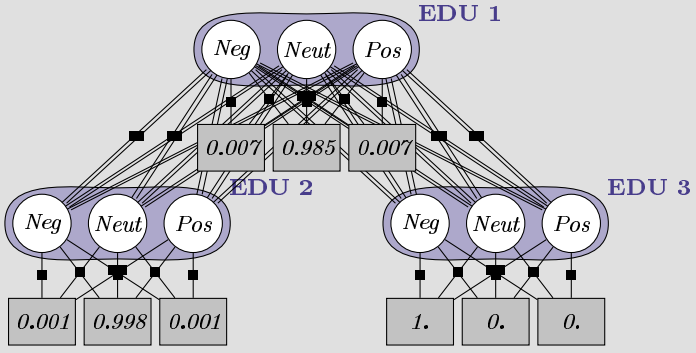
**Example 7.3.5 (Error Made by the ROOT System)**

**Tweet:** [Die NSA weiss auch von dir ...]$_1$ [Nützt uns auch nichts .]$_2$ [ %NegSmiley]$_3$

[*The NSA also knows about you*]$_1$ [*It doesn't help us either*]$_2$ [ *%NegSmiley*]$_3$

**Gold Label:** **negative**

**Predicted Label:** **neutral***




Finally, the last example (7.3.6) shows an error made by the LAST baseline system, which predicts the neutral label for a negative tweet based on the polarity of its right-most EDU. This unit indeed admits some positive moments with regard to the sad news expressed in the first segment, but in contrast to the movie description from Example 7.3.1, where the last

sentence completely overturned the polarity of the whole text, this time, the final opinion does not alter the general negative mood of the message, but only dampens its effect.

**Example 7.3.6 (Error Made by the LAST System)**
**Tweet:** [' ( :'( :'( Die letzte Aussprache war wohl das schwerste Telefonat meines gesamten Lebens :'( :'( :'(]$_1$ [Aber wir gehen friedlich und als F . . .]$_2$
[' ( :'( :'( *The last talk was probably the most difficult call in my entire life* :'( :'( :'(]$_1$ [*But we go apart peacefully and as f* . . .]$_1$

**Gold Label:** **negative**
**Predicted Label:** **neutral***

## 7.4 Evaluation

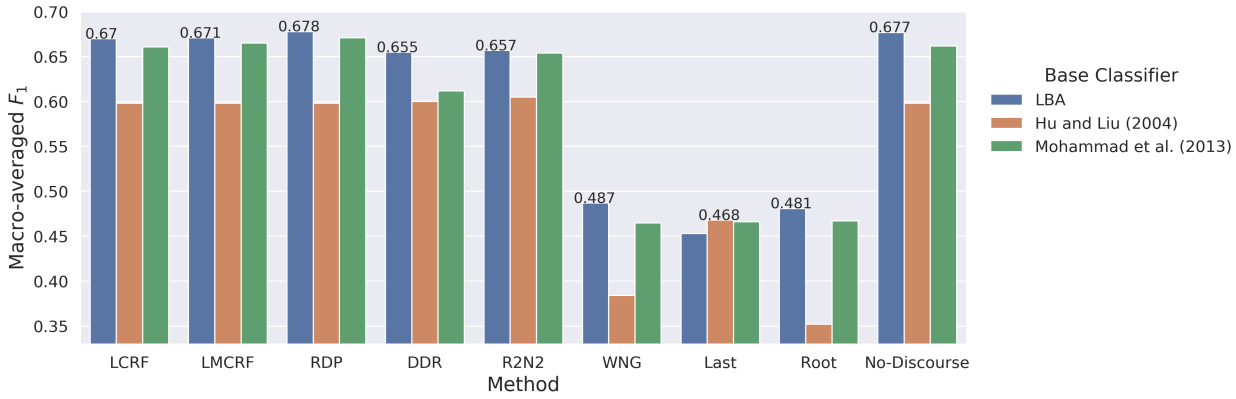
As we could see from the examples in Section 7.3, the results of our proposed methods were significantly limited by two key factors: (i) scores predicted by the base sentiment system for tweets and EDUs and (ii) structure of RST trees constructed for these messages. In order to estimate the effect of these factors more precisely, we decided to rerun our experiments, trying alternative solutions for each of these aspects.

### 7.4.1 Base Classifier

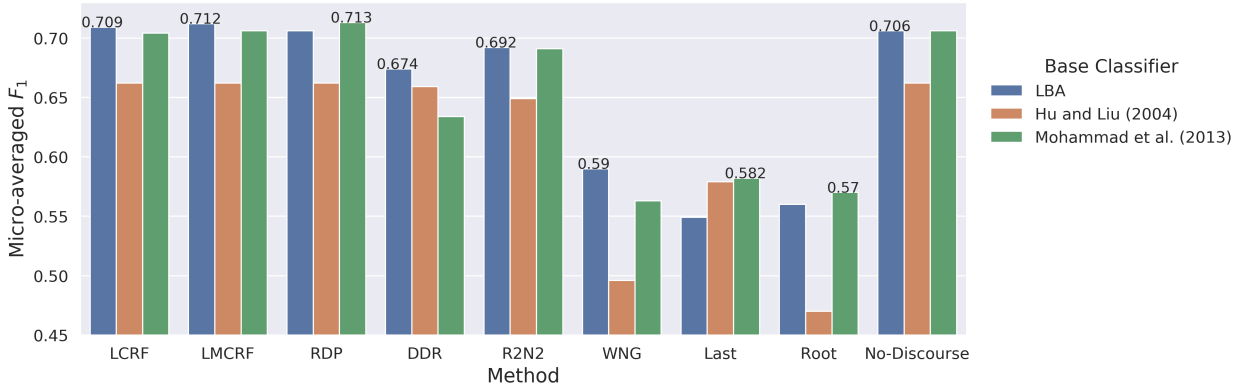
To assess the impact of the former factor (the quality of the base sentiment classifier), we replaced all polarity scores produced by the LBA system with the respective values predicted by the best lexicon- and ML-based CGSA methods (the systems of Hu and Liu [2004] and Mohammad et al. [2013] respectively) and retrained all DASA approaches on the updated data, subsequently evaluating them on the PotTS and SB10k test sets. The results of this evaluation are shown in Figures 7.10 and 7.11.

As we can see from the first figure, our initially chosen LBA approach is indeed a more amenable basis to almost all discourse-aware sentiment methods on the PotTS corpus. A few exceptions to this general rule are the macro-averaged $F_1$-score of the LAST baseline, which surprisingly improves in combination with the lexicon-based system, and the micro-average of the RDP and LAST methods, which attain their best results (0.713 and 0.582) in conjunction with the SVM classifier of Mohammad et al. (2013).

A slightly different situation is observed on the SB10k corpus though. On this dataset, LBA still leads to higher macro-$F_1$–scores for DDR, R2N2, WNG, LAST, and ROOT; but the approach of Mohammad et al. (2013) improves the results of LCRF, LMCRF, RDP, and




*(a) Macro-$F_1$*



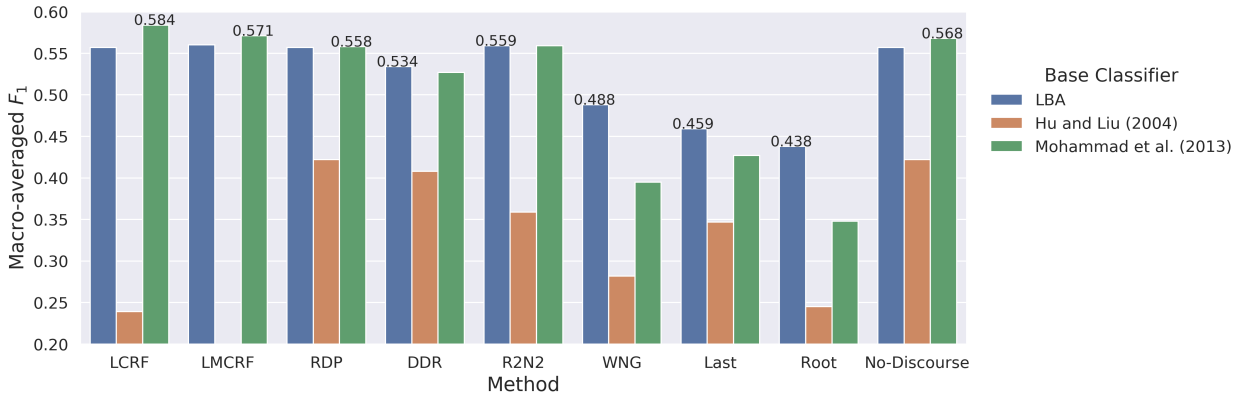

*(b) Micro-$F_1$*

*Figure 7.10: Results of discourse-aware sentiment analysis methods with different base classifiers on the PotTS corpus*

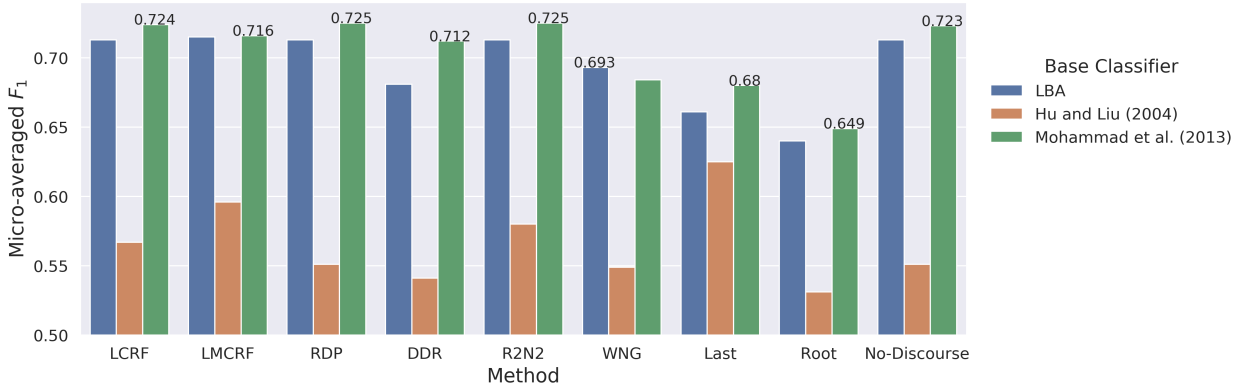
NO-DISCOURSE. The SVM classifier is also the unequivocal leader in terms of the micro-averaged $F_1$, yielding the highest scores for all systems except WNG. Unfortunately, the lexicon-based predictor of Hu and Liu (2004) performs much weaker than SVM and LBA: the highest macro- and micro-averaged $F_1$-scores achieved with this approach run up to 0.422 (RDP) and 0.625 (LAST) respectively. The most disappointing result for us, however, is that the LMCRF system completely fails to predict any polar class except NEUTRAL on the SB10k test set when trained with the scores of this method (see Figure 7.11a). Similarly, LCRF yields considerably lower scores in combination with this solution, reaching only 0.239 macro-$F_1$.

### 7.4.2 Parsing Quality and Relation Scheme

Another factor which could significantly influence the results of discourse-aware methods was the quality of the automatic RST parsing and the set of discourse relations distinguished by the parser system. Although improving the results of DPLP let alone manually annotating the complete PotTS and SB10k datasets was beyond the scope of our dissertation (even




*(a) Macro-$F_1$*




*(b) Micro-$F_1$*

*Figure 7.11: Results of discourse-aware sentiment analysis methods with different base classifiers on the SB10k corpus*

though we have made such attempt [see Sidarenka et al., 2015a]), we decided to check whether at least evaluating the DASA methods on manually annotated data would improve their results. For this purpose, we asked a student assistant to segment and parse 88% of the tweets from the PotTS test set[11] and tested all DASA approaches on these hand-crafted RST data.

As we can see from the results in Table 7.2, the scores of all systems except WNG, LAST, and ROOT increase by three to four percent. Even the macro-averaged $F_1$-measure of the discourse-unaware classifier improves from 0.677 to 0.716, as does its micro-$F_1$–score, which rises from 0.706 to 0.753 $F_1$. These last changes, however, are exclusively due to the reduced size of the test set, since the discourse-unaware method does not take RST trees into account. Unfortunately, this time, NO-DISCOURSE also outperforms all discourse-aware approaches in terms of the micro-averaged $F_1$, achieving an accuracy of 75,3%, although it still loses to the Recursive Dirichlet Process on the macro-averaged metric, yielding a 0.2% worse result than RDP (0.716 versus 0.718 macro-$F_1$). Another surprising finding for us is that in gold discourse annotation, EDUs which determine the actual polarity of the tweet are unlikely

[11]Unfortunately, due to the limited availability of the student, we could not annotate the whole test set.

| Method | Positive | | | Negative | | | Neutral | | | Macro $F_1$ | Micro $F_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Precision | Recall | $F_1$ | Precision | Recall | $F_1$ | Precision | Recall | $F_1$ | | |
| PotTS | | | | | | | | | | | |
| LCRF | 0.82 | 0.82 | **0.82** | **0.66** | 0.55 | 0.6 | 0.69 | 0.75 | 0.72 | 0.71 | 0.747 |
| LMCRF | **0.83** | 0.81 | **0.82** | 0.65 | 0.55 | 0.6 | 0.69 | **0.78** | **0.73** | 0.709 | 0.749 |
| RDP | 0.8 | 0.84 | **0.82** | 0.64 | 0.58 | 0.61 | **0.72** | 0.71 | 0.72 | **0.718** | 0.751 |
| DDR | 0.78 | 0.75 | 0.77 | 0.58 | **0.66** | **0.62** | 0.66 | 0.63 | 0.64 | 0.693 | 0.698 |
| R2N2 | 0.81 | 0.82 | 0.81 | 0.64 | 0.53 | 0.58 | 0.68 | 0.74 | 0.71 | 0.697 | 0.737 |
| WNG | 0.58 | 0.74 | 0.65 | 0.63 | 0.19 | 0.29 | 0.51 | 0.51 | 0.51 | 0.47 | 0.558 |
| LAST | 0.55 | **0.86** | 0.67 | 0.51 | 0.11 | 0.18 | 0.56 | 0.35 | 0.43 | 0.426 | 0.55 |
| ROOT | 0.58 | 0.56 | 0.57 | 0.58 | 0.25 | 0.35 | 0.43 | 0.6 | 0.5 | 0.46 | 0.513 |
| NO-DISCOURSE | 0.81 | 0.84 | **0.82** | 0.65 | 0.57 | 0.61 | **0.72** | 0.73 | **0.73** | 0.716 | **0.753** |

*Table 7.2: Results of discourse-aware sentiment analysis methods on the PotTS corpus with manually annotated RST trees*

to appear either at the end of a message or at the top of its RST tree, which explains the degradation of the scores for the LAST and ROOT baselines.

A common way to improve the quality of automatic RST parsing and ease the task of discourse-aware sentiment methods is to reduce the number of discourse relations distinguished by the parser system. Drawing on the work of Bhatia et al. (2015), we also used this approach, projecting all coherence links from the Potsdam Commentary Corpus (Stede and Neumann, 2014) to the binary set of CONTRASTIVE and NON-CONTRASTIVE ones. Although similar approximations were made in almost all other discourse-aware solutions (cf. Chenlo et al., 2013; Heerschop et al., 2011; Zhou et al., 2011), we were not sure whether the subset that we used was indeed optimal and sufficient to reflect all possible discourse interactions that could play an important role in sentiment composition.

To answer this question, we retrained the DPLP parser on the PCC, using the subsets of relations proposed by Chenlo et al. (2013), Heerschop et al. (2011), and Zhou et al. (2011), and also tried the original set of all RST links from the Potsdam Commentary Corpus. A detailed overview of these sets is given in Table 7.3.

To check whether cardinalities of these sets indeed correlate with the quality of automatic RST parsing, we evaluated each retrained system on the held-out PCC test data and present the results of this evaluation in Table 7.4. As is evident from the scores, coarser relation schemes in fact improve parsing quality, especially in terms of relation $F_1$. In the most extreme case (*e.g.*, Bhatia et al. [which has only two links] versus PCC [which comprises 34 relations]), these gains can reach up to seven percent. However, with respect to other metrics (span and nuclearity $F_1$), the gaps are notably smaller and might even be in favor of the richer relation set (cf. nuclearity $F_1$ for PCC).

To see how this varying quality affected the net results of discourse-aware sentiment

| Scheme | Relation Set | Equivalence Classes |
|---|---|---|
| Bhatia et al. | {Contrastive, **Non-Contrastive**} | Contrastive := {Antithesis, Antithesis-E, Comparison, Concession, Consequence-S, Contrast, Problem-Solution}. |
| Chenlo et al. | {Attribution, Background, Cause, Comparison, Condition, Consequence, Contrast, Elaboration, Enablement, Evaluation, Explanation, Joint, Otherwise, Temporal, **Other**} | |
| Heerschop et al. | {Attribution, Background, Cause, Condition, Contrast, Elaboration, Enablement, Explanation, **Other**} | |
| PCC | {Antithesis, Background, Cause, Circumstance, Concession, Condition, Conjunction, Contrast, Disjunction, E-Elaboration, Elaboration, Enablement, Evaluation-N, Evaluation-S, Evidence, Interpretation, Joint, Justify, List, Means, Motivation, Otherwise, Preparation, Purpose, Reason, Restatement, Restatement-MN, Result, Sequence, Solutionhood, Summary, Unconditional, Unless, Unstated-Relation} | |
| Zhou et al. | {Contrast, Condition, Continuation, Cause, Purpose, **Other**} | Contrast := {Antithesis, Concession, Contrast, Otherwise}; Continuation := {Continuation, Parallel}; Cause := {Evidence, Nonvolitional-Cause, Nonvolitional-Result, Volitional Cause, Volitional-Result}; |

*Table 7.3: RST relations used in the original Potsdam Commentary Corpus and different discourse-aware sentiment methods (default relation [which subsumes the rest of the links] is shown in **boldface**)*

methods, we re-evaluated all DASA approaches on the updated automatic RST trees and show the results of this evaluation in Figures 7.12 and 7.13.

As it turns out, latent-marginalized CRF can still hold the overall record in both macro- and micro-averaged $F_1$ on the PotTS corpus, although its margin to the closest competitor (R2N2) is relatively small, amounting to only 0.1 percent. Interestingly enough, both top-performing methods (LMCRF and R2N2) achieve their best results with richer relation sets than the one we used in our initial experiment: For example, LMCRF attains its highest macro-score in combination with the relation scheme of Heerschop et al. (2011) and yields the best micro-$F_1$ when used with the scheme of Chenlo et al. (2014). The rhetorical recursive neural network, vice versa, attains its highest macro-average with the latter relation set and reaches its best micro-$F_1$ in conjunction with the former subset of relations.

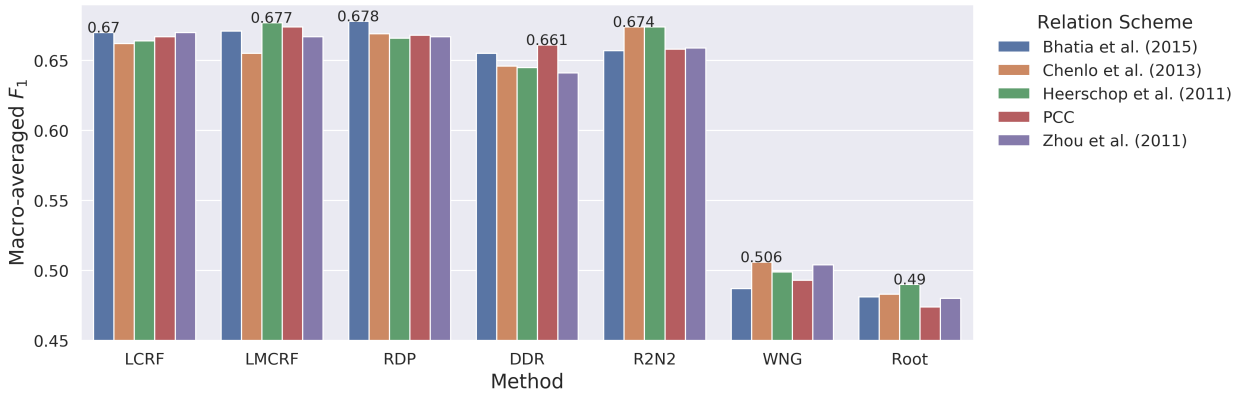
A different situation is observed with other DASA approaches though. For example, LCRF and RDP perform best when used with the initially chosen set of Bhatia et al. (2015).

| Relation Scheme | Span $F_1$ | Nuclearity $F_1$ | Relation $F_1$ |
|---|---|---|---|
| Bhatia et al. | **0.777** | 0.512 | **0.396** |
| Chenlo et al. | 0.769 | 0.505 | 0.362 |
| Heerschop et al. | 0.774 | 0.51 | 0.361 |
| PCC | 0.776 | **0.534** | 0.326 |
| Zhou et al. | 0.776 | 0.501 | 0.388 |

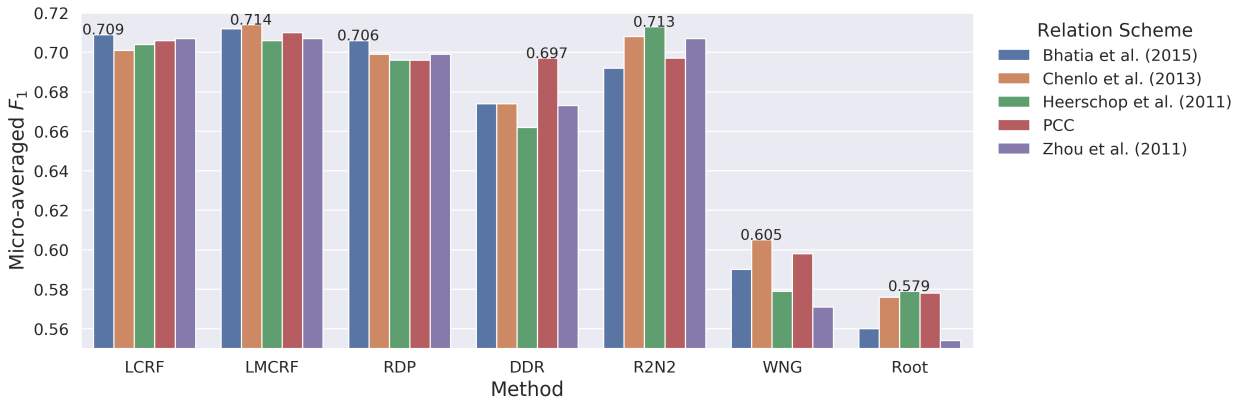
*Table 7.4: Results of the DPLP parser on PCC 2.0 with different relation schemes*

On the other hand, discourse-depth reweighting strongly benefits from the full unconstrained PCC relations, which is probably due to the better nuclearity classification achieved with this scheme. Finally, WNG and ROOT reach their best results with the relation subsets proposed by Chenlo et al. and Heerschop et al. respectively.

A much more uniform situation is observed on the SB10k corpus (see Figure 7.13), where the $F_1$-scores of our methods vary only slightly across different relation schemes. The only significant improvements that we can notice this time are higher macro- and micro-averaged $F_1$s achieved by the RDP approach in combination with the Heerschop et al.'s subset. This

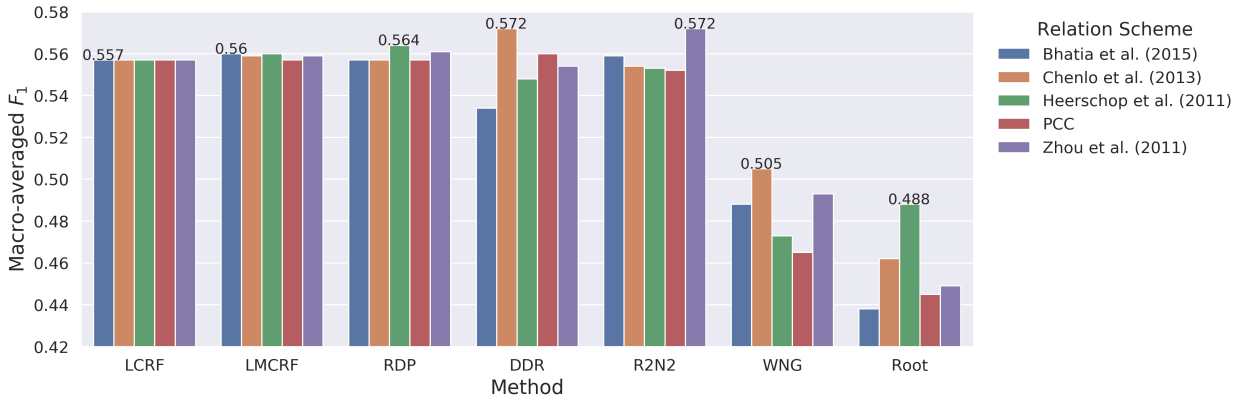



*(a) Macro-$F_1$*

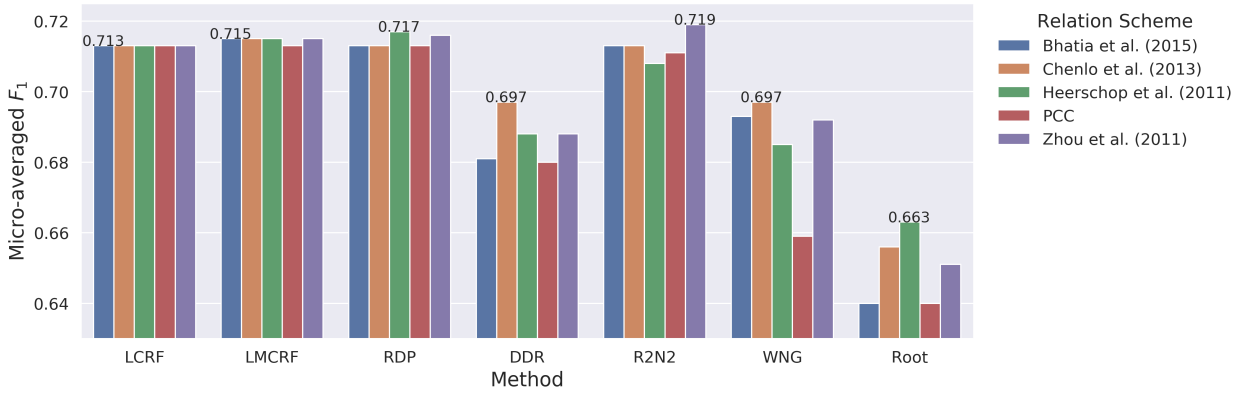



*(b) Micro-$F_1$*

*Figure 7.12: Results of discourse-aware sentiment classifiers for different relation schemes on the PotTS corpus*




*(a)* `Macro-`$F_1$




*(b)* `Micro-`$F_1$

*Figure 7.13: Results of discourse-aware sentiment classifiers for different relation schemes on the SB10k corpus*

subset is also most amenable to the ROOT baseline, which reaches 0.488 macro-$F_1$ and 0.663 micro-$F_1$, significantly improving on its initial results. At the same time, discourse-depth reweighting and the approach of Wang and Wu capitalize on the relations defined by Chenlo et al. so much that the former system even achieves the highest overall macro-$F_1$–score (0.572), being on a par with the R2N2 system.

## 7.5 Summary and Conclusions

At this point, our chapter has come to an end and, concluding it, we would like to recap that in this part of the thesis:

- we have presented an overview of the most popular approaches to automatic discourse analysis (RST, PDTB, and SDRT) and explained why we think that one of these frameworks (Rhetorical Structure Theory) would be more amenable to the purposes of discourse-aware sentiment analysis than the others;

- to substantiate our claims and to see whether the lexicon-based attention system introduced in the previous chapter would indeed benefit from awareness of discourse structure, we segmented all microblogs from the PotTS and SB10k corpora into elementary discourse units using the SVM-based segmenter of Sidarenka et al. (2015b) and parsed these messages with the RST parser of Ji and Eisenstein (2014), which had been previously retrained on the Potsdam Commentary Corpus (Stede and Neumann, 2014);

- afterwards, we estimated the results of existing discourse-aware sentiment methods (the systems of Wang et al. [2015b] and Bhatia et al. [2015]) and also evaluated two simpler baselines (in which we predicted semantic orientation of a tweet by taking the polarity of its last and root EDUs), getting the best results with the R2N2 solution of Bhatia et al. (2015) (0.657 and 0.559 macro-$F_1$ on PotTS and SB10k respectively);

- we could, however, improve on these scores and also outperform the plain LBA system with our three proposed discourse-aware sentiment solutions (latent and latent-marginalized conditional random fields and Recursive Dirichlet Process), boosting the macro-averaged $F_1$-score on PotTS to 0.678 and increasing the result on SB10k to 0.56 macro-$F_1$;

- a subsequent evaluation of these approaches with different settings showed that the results of all discourse-aware methods largely correlated with the scores of the base sentiment classifier and also revealed an important drawback of the latent-marginalized CRFs, which failed to predict any positive or negative instance on the test set of the SB10k corpus when trained in combination with the lexicon-based approach of Hu and Liu (2004);

- nevertheless, almost all DASA solutions could improve their scores when tested on manually annotated RST trees or used with a richer set of discourse relations.