SciDTB: Discourse Dependency TreeBank for Scientific Abstracts

An Yang Sujian Li*
Key Laboratory of Computational Linguistics, Peking University, MOE, China
{yangan, lisujian}@pku.edu.cn

Abstract

Annotation corpus for discourse relations benefits NLP tasks such as machine translation and question answering. In this paper, we present SciDTB, a domain-specific discourse treebank annotated on scientific articles. Different from widely-used RST-DT and PDTB, SciDTB uses dependency trees to represent discourse structure, which is flexible and simplified to some extent but do not sacrifice structural integrity. We discuss the labeling framework, annotation workflow and some statistics about SciDTB. Furthermore, our treebank is made as a benchmark for evaluating discourse dependency parsers, on which we provide several baselines as fundamental work.

1 Introduction

Discourse relation depicts how the text spans in a text relate to each other. These relations can be categorized into different types according to semantics, logic or writer's intention. Annotations of such discourse relations can benefit many down-stream NLP tasks including machine translation (Guzmán et al., 2014; Joty et al., 2014) and automatic summarization (Gerani et al., 2014).

Several discourse corpora have been proposed in previous work, grounded with various discourse theories. Among them Rhetorical Structure Theory TreeBank (RST-DT) (Carlson et al., 2003) and Penn Discourse TreeBank (PDTB) (Prasad et al., 2007) are the most widely-used resources. PDTB focuses on shallow discourse relations between two arguments and ignores the whole organization. RST-DT based on Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) represents a text into a hierarchical discourse tree. Though RST-DT provides more comprehensive discourse information, its limitations including the introduction of intermediate nodes and absence of non-projective structures bring the annotation and parsing complexity.

Li et al. (2014) and Yoshida et al. (2014) both realized the problems of RST-DT and introduced dependency structures into discourse representation. Stede et al. (2016) adopted dependency tree format to compare RST structure and Segmented Discourse Representation Theory(SDRT) (Lascarides and Asher, 2008) structure for a corpus of short texts. Their discourse dependency framework is adapted from syntactic dependency structure (Hudson, 1984; Böhmová et al., 2003), with words replaced by elementary discourse units (EDUs). Binary discourse relations are represented from dominant EDU (called "head") to subordinate EDU (called "dependent"), which makes non-projective structure possible. However, Li et al. (2014) and Yoshida et al. (2014) mainly focused on the definition of discourse dependency structure and directly transformed constituency trees in RST-DT into dependency trees. On the one hand, they only simply treated the transformation ambiguity, while constituency structures and dependency structures did not correspond one-to-one. On the other hand, the transformed corpus still did not contain non-projective dependency trees, though "crossed dependencies" actually exist in the real flexible discourse structures (Wolf and Gibson, 2005). In such case, it is essential to construct a discourse dependency treebank from scratch instead of through automatically converting from the constituency structures.

In this paper, we construct the discourse dependency corpus SciDTB[1]. based on scientific abstracts, with the reference to the discourse de-

[1] The treebank is available at https://github.com/PKU-TANGENT/SciDTB

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 444–449
Melbourne, Australia, July 15 - 20, 2018. ©2018 Association for Computational Linguistics

pendency representation in Li et al. (2014). We choose scientific abstracts as the corpus for two reasons. First, we observe that when long news articles in RST-DT have several paragraphs, the discourse relations between paragraphs are very loose and their annotations are not so meaningful. Thus, short texts with obvious logics become our preference. Here, we choose scientific abstracts from ACL Anthology[2] which are usually composed of one passage and have strong logics. Second, we prefer to conduct domain-specific discourse annotation. RST-DT and PDTB are both constructed on news articles, which are unspecific in domain coverage. We choose the scientific domain that is more specific and can benefit further academic applications such as automatic summarization and translation. Furthermore, our treebank SciDTB can be made as a benchmark for evaluating discourse parsers. Three baselines are provided as fundamental work.

2 Annotation Framework

In this section, we describe two key aspects of our annotation framework, including elementary discourse units (EDU) and discourse relations.

2.1 Elementary Discourse Units

We first need to divide a passage into non-overlapping text spans, which are named elementary discourse units (EDUs). We follow the criterion of Polanyi (1988) and Irmer (2011) which treats clauses as EDUs.

However, since a discourse unit is a semantic concept but a clause is defined syntactically, in some cases segmentation by clauses is still not the most proper strategy. In practice, we refer to the guidelines defined by (Carlson and Marcu, 2001). For example, subjective clauses, objective clauses of non-attributive verbs and verb complement clauses are not segmented. Nominal postmodifiers with predicates are treated as EDUs. Strong discourse cues such as "*despite*" and "*because of*" starts a new EDU no matter they are followed by a clause or a phrase. We give an EDU segmentation example as follows.

1. *[Despite bilingual embeddings success,][**the contextual information**][which is of critical importance to translation quality,][**was ignored in previous work.**]*

	Coarse	Fine
1.	ROOT	ROOT
2.	Attribution	Attribution
3.	Background	Related, Goal, General
4.	Cause-effect	Cause, Result
5.	Comparison	Comparison
6.	Condition	Condition
7.	Contrast	Contrast
8.	Elaboration	Addition, Aspect, Process-step, Definition, Enumerate, Example
9.	Enablement	Enablement
10.	Evaluation	Evaluation
11.	Explain	Evidence, Reason
12.	Joint	Joint
13.	Manner-means	Manner-means
14.	Progression	Progression
15.	Same-unit	Same-unit
16.	Summary	Summary
17.	Temporal	Temporal

Table 1: Discourse relation category of SciDTB.

It is noted, as in Example 1, there are EDUs which are broken into two parts (in bold) by relative clauses or nominal postmodifiers. Like RST, we connect the two parts by a pseudo-relation type *Same-unit* to represent their integrity.

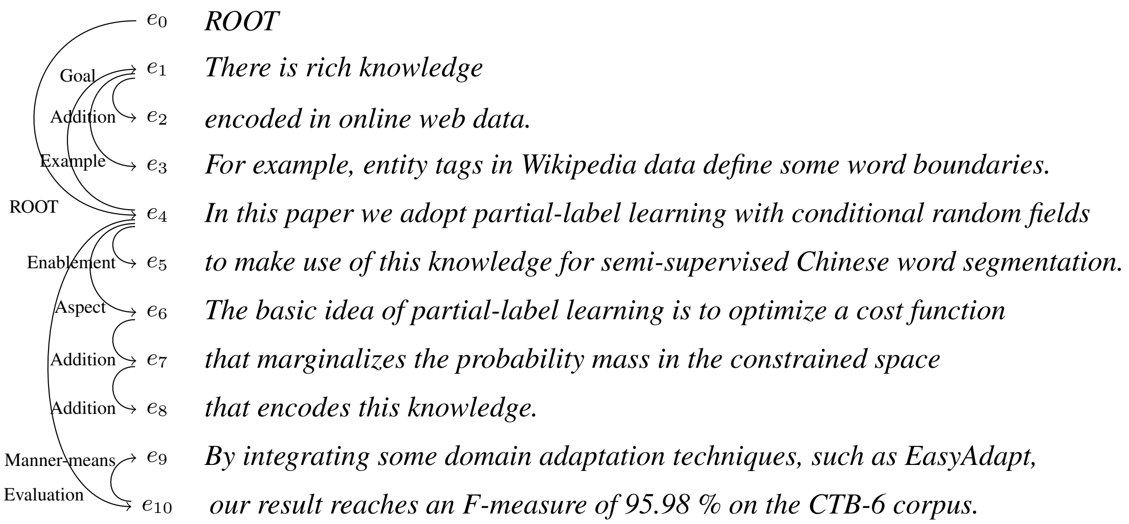
2.2 Discourse Relations

A discourse relation is defined as tri-tuple (h, d, r), where h means the head EDU, d is the dependent EDU, and r defines the relation category between h and d. For a discourse relation, head EDU is defined as the unit with essential information and dependent EDU with supportive content. Here, we follow Carlson and Marcu (2001) to adopt deletion test in the determination of head and dependent. If one unit in a binary relation pair is deleted but the whole meaning can still be almost understood from the other unit, the deleted unit is treated as dependent and the other one as the head.

For the relation categories, we mainly refer to the work of (Carlson and Marcu, 2001) and (Bunt and Prasad, 2016). Table 1 presents the discourse relation set of SciDTB, which are not explained detailedly one by one due to space limitation. Through investigation of scientific abstracts, we define 17 coarse-grained relation types and 26 fine-grained relations for SciDTB.

It is noted that we make some modifications to adapt to the scientific domain. For example, In SciDTB, *Background* relation is divided into three

[2] http://www.aclweb.org/anthology/. SciDTB follows the same CC BY-NC-SA 3.0 and CC BY 4.0 licenses as ACL Anthology.



e_0 *ROOT*
e_1 *There is rich knowledge*
e_2 *encoded in online web data.*
e_3 *For example, entity tags in Wikipedia data define some word boundaries.*
e_4 *In this paper we adopt partial-label learning with conditional random fields*
e_5 *to make use of this knowledge for semi-supervised Chinese word segmentation.*
e_6 *The basic idea of partial-label learning is to optimize a cost function*
e_7 *that marginalizes the probability mass in the constrained space*
e_8 *that encodes this knowledge.*
e_9 *By integrating some domain adaptation techniques, such as EasyAdapt,*
e_{10} *our result reaches an F-measure of 95.98 % on the CTB-6 corpus.*

Figure 1: An example discourse dependency tree for an abstract in SciDTB.

subtypes: *Related*, *Goal* and *General*, because the background description in scientific abstracts usually has more different intents. Meanwhile, for *attribution* relation we treat the attributive content rather than act as head, which is contrary to that defined in (Carlson and Marcu, 2001), because scientific facts or research arguments mentioned in attributive content are more important in abstracts. For some symmetric discourse relations such as *joint* and *comparison*, where two connected EDUs are equally important and have interchangeable semantic roles, we follow the strategy as (Li et al., 2014) and treat the preceding EDU as the head.

Another issue on coherence relations is about polynary relations which involve more than two EDUs. The first scenario is that one EDU dominates a set of posterior EDUs as its member. In this case, we annotate binary relations from head EDU to each member EDU with the same relation. The second scenario is that several EDUs are of equal importance in a polynary relation. For this case, we link each former EDU to its neighboring EDU with the same relation, forming a relation chain similar to "right-heavy" binarization transformation in (Morey et al., 2017).

By assuring that each EDU has one and only one head EDU, we can obtain a dependency tree for each scientific abstract. An example of dependency annotation is shown in Figure 1.

3 Corpus Construction

Following the annotation framework, we collected 798 abstracts from ACL anthology and constructed the SciDTB corpus. The construction details are introduced as follows.

Annotator Recruitment To select annotators, we put forward two requirements to ensure the annotation quality. First, we required the candidates to have linguistic knowledge. Second, each candidate was asked to join a test annotation of 20 abstracts, whose quality was evaluated by experts. After the judgement, 5 annotators were qualified to participate in our work.

EDU Segmentation We performed EDU segmentation in a semi-automatic way. First, we did sentence tokenization on raw texts using NLTK 3.2 (Bird and Loper, 2004). Then we used SPADE (Soricut and Marcu, 2003), a pre-trained EDU segmenter relying on Charniak's syntactic parser (Charniak, 2000), to automatically cut sentences into EDUs. Then, we manually checked each segmented abstract to ensure the segmentation quality. Two annotators conducted the checking task, with one proofreading the output of SPADE, and the other reviewing the proofreading. The checking process was recorded for statistical analysis.

Tree Annotation Labeling dependency trees was the most labor-intensive work in the corpus construction. 798 segmented abstracts were labeled by 5 annotators in 6 months. 506 abstracts were annotated more than twice separately by different annotators, with the purpose of analysing annotation consistency and providing human performance as an upper bound. The annotated trees were stored in JSON format. For convenience, we developed an online tool[3] for annotating and visualising discourse dependency trees.

4 Corpus Statistics

SciDTB contains 798 unique abstracts with 63% labeled more than once and 18,978 discourse relations in total. Table 2 compares the size of SciDTB with RST-DT and another PDTB-style domain-specific corpus BioDRB (Prasad et al., 2011), we can see SciDTB has a comparable size with RST-DT. Moreover, it is relatively easy for SciDTB to augment its size since the dependency structure simplifies the annotation to some extent. Compared with BioDRB, SciDTB has larger size and passage-level representations.

Corpus	#Doc.	#Doc. (unique)	#Relation
SciDTB	1355	798	18978
RST-DT	438	385	23611
BioDRB	24	24	5859

Table 2: Size of SciDTB and other discourse relation banks.

4.1 Annotation Consistency

EDU Segmentation We use 214 abstracts for analysis. After the proofreading of the first annotator, the abstracts are composed of totally 2,772 EDUs. Among these EDUs, only 28 (1.01%) EDUs are disagreed and revised by the second annotator, which means a very high consensus between annotators on EDU segmentation.

Annotator	#Doc.	UAS	LAS	Kappa score
1 & 2	93	0.811	0.644	0.763
1 & 3	147	0.800	0.628	0.761
1 & 4	42	0.772	0.609	0.767
3 & 4	46	0.806	0.639	0.772
4 & 5	44	0.753	0.550	0.699

Table 3: Relation annotation consistency.

Tree Labeling Here, we evaluate the consistency of two annotators on labeling discourse relations using 3 metrics from different aspects. When labeling a discourse relation, each non-root EDU must choose its head with a specific relation type. Thus, the annotation disagreement mainly comes from selecting head or determining relation type. Similar to syntactic dependency parsing, unlabeled and labeled attachment scores (**UAS** and **LAS**) are employed to measure the labeling correspondence. UAS calculates the proportion of EDUs which are assigned the same head in two annotations, while LAS considers the uniformity of both head and relation label. **Cohen's Kappa score** evaluates the agreement of labeling relation types under the premise of knowing the correct heads.

Distance	#Relations	Percentage/%
0 EDU	10576	61.64
1 EDU	2208	12.87
2 EDUs	1231	7.17
3-5 EDUs	1626	9.48
6-10 EDUs	1146	6.68
11-15 EDUs	304	1.77
>15 EDUs	67	0.39
Total	17158	100.00

Table 4: Distribution of dependency distance.

Table 3 shows the agreement results between two annotators. We can see that most of the **LAS** values between annotators exceed 0.60. The agreement on tree structure reflected by **UAS** all reaches 0.75. The **Kappa** values for relation types agreement keep equal to or greater than 0.7.

4.2 Structural Characteristics

Non-projection in Treebank One advantage of dependency trees is that they can represent non-projective structures. In SciDTB, we annotated 39 non-projective dependency trees, which account for about 3% of the whole corpus. This phenomenon in our treebank is not so frequent as (Wolf and Gibson, 2005). We think this may be because scientific abstracts are much shorter and scientific expressions are relatively restricted.

Dependency Distance Here we investigate the distance of two EDUs involved in a discourse relation. The distance is defined as the number of EDUs between head and dependent. We present the distance distribution of all the relations in SciDTB, as shown in Table 4. It should be noted that *ROOT* and *Same-unit* relations are omitted in this analysis. From Table 4, we find most relations connect near EDUs. Most relations (61.6%) occur between neighboring EDUs and about 75% relations occur with at most one intermediate EDU.

Although most dependency relations function intra-sentence, there exist long-range dependency relations in the treebank. On average, the distance of 8.8% relations is greater than 5. We summarize that the most frequent 5 fine-grained rela-

[3] http://123.56.88.210/demo/depannotate/

tion types of these long-distance relations belong to *Evaluation*, *Aspect*, *Addition*, *Process-step* and *Goal*, which tend to appear on higher level in dependency trees.

5 Benchmark for Discourse Parsers

We further apply SciDTB as a benchmark for comparing and evaluating discourse dependency parsers. For the 798 unique abstracts in SciDTB, 154 are used for development set and 152 for test set. The remaining 492 abstracts are used for training. We implement two transition-based parsers and a graph-based parser as baselines.

Vanilla Transition-based Parser We adopt the transition-based method for dependency parsing by Nivre (2003). The action set of arc-standard system (Nivre et al., 2004) is employed. We build an SVM classifier to predict most possible transition action for given configuration. We adopt the N-gram features, positional features, length features and dependency features for top-2 EDUs in the stack and top EDU in the buffer, which can be referred from (Li et al., 2014; Wang et al., 2017)

Two-stage Transition-based Parser We implement a two-stage transition-based dependency parser following (Wang et al., 2017). First, an unlabeled tree is produced by vanilla transition-based approach. Then we train a separate SVM classifier to predict relation types on the tree in pre-order. For the 2nd-stage, apart from features in the 1st-stage, two kinds of features are added, including depth of head and dependent in the tree and the predicted relation between the head and its head.

Graph-based Parser We implement a graph-based parser as in (Li et al., 2014). For simplicity, we use averaged perceptron rather than MIRA to train weights. N-gram, positional, length and dependency features between head and dependent labeled with relation type are considered.

Hyper-parameters During training, the hyper-parameters of these models are tuned using development set. For vanilla transition-based parser, we take linear kernel for the SVM classifier. The penalty parameter C is set to 1.5. For two-stage parser, the 1st-stage classifier follows the same setting as the vanilla parser. For 2nd-stage, we use the linear kernel and set C to 0.5. The averaged perceptron in graph-based parser is trained for 10 epochs on the training set. Weights of features are initialized to be 0 and trained with fixed learning rate.

	Dev set		Test set	
	UAS	LAS	UAS	LAS
Vanilla transition	**0.730**	0.557	**0.702**	0.535
Two-stage transition	**0.730**	**0.577**	**0.702**	**0.545**
Graph-based	0.607	0.455	0.576	0.425
Human	0.806	0.627	0.802	0.622

Table 5: Performance of baseline parsers.

Results Table 5 shows the performance of these parsers on development and test data. We also measure parsing accuracy with UAS and LAS. The human agreement is presented for comparison. With the addition of tree structural features in relation type prediction, the two-stage dependency parser gets better performance on LAS than vanilla system on both development and test set. Compared with graph-based model, the two transition-based baselines achieve higher accuracy with regard to UAS and LAS. Using more effective training strategies like MIRA may improve graph-based models. We can also see that human performance is still much higher than the three parsers, meaning there is large space for improvement in future work.

6 Conclusions

In this paper, we propose to construct a discourse dependency treebank SciDTB for scientific abstracts. It represents passages with dependency tree structure, which is simpler and more flexible for analysis. We have presented our annotation framework, construction workflow and statistics of SciDTB, which can provide annotation experience for extending to other domains. Moreover, this treebank can serve as an evaluating benchmark of discourse parsers.

In the future, we will enlarge our annotation scale to cover more domains and longer passages, and explore how to use SciDTB in some down-streaming applications.

Acknowledgments

We thank the anonymous reviewers for their insightful comments on this paper. We especially thank Liang Wang for developing the online annotation tool. This work was partially supported by National Natural Science Foundation of China (61572049 and 61333018). The correspondence author of this paper is Sujian Li.

References

Steven Bird and Edward Loper. 2004. Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. Association for Computational Linguistics, page 31.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2003. The prague dependency treebank. In *Treebanks*, Springer, pages 103–127.

Harry Bunt and Rashmi Prasad. 2016. Iso dr-core (iso 24617-8): Core concepts for the annotation of discourse relations. In *Proceedings 12th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-12)*. pages 45–54.

Lynn Carlson and Daniel Marcu. 2001. Discourse tagging reference manual. *ISI Technical Report ISI-TR-545* 54:56.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*, Springer, pages 85–112.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*. Association for Computational Linguistics, pages 132–139.

Shima Gerani, Yashar Mehdad, Giuseppe Carenini, Raymond T Ng, and Bita Nejat. 2014. Abstractive summarization of product reviews using discourse structure. In *EMNLP*. volume 14, pages 1602–1613.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2014. Using discourse structure improves machine translation evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. volume 1, pages 687–698.

Richard A Hudson. 1984. *Word grammar*. Blackwell Oxford.

Matthias Irmer. 2011. *Bridging inferences: Constraining and resolving underspecification in discourse interpretation*, volume 11. Walter de Gruyter.

Shafiq Joty, Francisco Guzmán, Lluís Màrquez, and Preslav Nakov. 2014. Discotk: Using discourse structure for machine translation evaluation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*. pages 402–408.

Alex Lascarides and Nicholas Asher. 2008. Segmented discourse representation theory: Dynamic semantics with discourse structure. In *Computing meaning*, Springer, pages 87–124.

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)*. volume 1, pages 25–35.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse* 8(3):243–281.

Mathieu Morey, Philippe Muller, and Nicholas Asher. 2017. How much progress have we made on rst discourse parsing? a replication study of recent results on the rst-dt. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pages 1319–1324.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT*. Citeseer.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2004. Memory-based dependency parsing. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*.

Livia Polanyi. 1988. A formal model of the structure of discourse. *Journal of pragmatics* 12(5-6):601–638.

Rashmi Prasad, Susan McRoy, Nadya Frid, Aravind Joshi, and Hong Yu. 2011. The biomedical discourse relation bank. *BMC bioinformatics* 12(1):188.

Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie L Webber. 2007. The penn discourse treebank 2.0 annotation manual .

Radu Soricut and Daniel 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*. Association for Computational Linguistics, pages 149–156.

Manfred Stede, Stergos D Afantenos, Andreas Peldszus, Nicholas Asher, and Jérémy Perret. 2016. Parallel discourse annotations on a corpus of short texts. In *LREC*.

Yizhong Wang, Sujian Li, and Houfeng Wang. 2017. A two-stage parsing method for text-level discourse analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. volume 2, pages 184–188.

Florian Wolf and Edward Gibson. 2005. Representing discourse coherence: A corpus-based study. *Computational Linguistics* 31(2):249–287.

Yasuhisa Yoshida, Jun Suzuki, Tsutomu Hirao, and Masaaki Nagata. 2014. Dependency-based discourse parser for single-document summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pages 1834–1839.