

A Constituent Syntactic Parse Tree Based Discourse Parser

Zhongyi Li^{1,2}, Hai Zhao^{1,2,*}, Chenxi Pang^{1,2}, Lili Wang^{1,2}, Huan Wang³

¹Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai, 200240, China

²Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

³Omron Institute of Sensing & Control Technology(Shanghai)

{rival2710, wang_lili}@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn
pcx0558@163.com, hwang8@gc.omron.com

Abstract

This paper describes our system in the CoNLL-2016 shared task. Our system takes a piece of newswire text as input and returns the discourse relations. In our system we use a pipeline to conduct each subtask. Our system is evaluated on the CoNLL-2016 Shared Task closed track and obtains 0.1515 in F1 measurement, especially the part of detecting connectives, which achieves 0.9838 on blind test set.

1 Introduction

An end-to-end discourse parser is a system using natural language text as input and the discourse relation in labeled text as output. It has been widely used in the field of natural language processing, such as text classification, question answering system. In these discourse relations, two argument spans are marked as targets looked for by discourse relations, while conjunctions (connective) play an important role to confirm the relationship between the two argument spans. According to whether the conjunctions clearly appear in the text, discourse relations can be divided into two categories: explicit and non-explicit.

Penn Discourse Treebank (PDTB) has become the most important corpus in the field of discourse parsing. Previous work (Lin et al., 2014) integrated the entire training process together to form

*Corresponding author. This paper was partially supported by Cai Yuanpei Program (CSC No. 201304490199 and No. 201304490171), National Natural Science Foundation of China (No. 61170114 and No. 61272248), National Basic Research Program of China (No. 2013CB329401), Major Basic Research Program of Shanghai Science and Technology Committee (No. 15JC1400103), Art and Science Interdisciplinary Funds of Shanghai Jiao Tong University (No. 14JCRZ04), and Key Project of National Society Science Foundation of China (No. 15-ZDA041).

a complete discourse parser. There were five major components in the system, including Connective classifier, Argument labeler, Explicit classifier, Non-Explicit classifier, Attribution span labeler, with a part of PDTB as the training, which has achieved good prediction performance.

2 System Overview

We design our discourse parser as a sequential pipeline, shown in Figure 1. The whole system can be divided into two main parts: explicit and non-explicit.

The Explicit part contains:

(1) **Connective Classifier** Detects the discourse connectives. Note that not all commonly used conjunctions have the effect of connective, so we first identify the ones function as discourse connective.

(2) **Explicit Argument Labeler** Locates the relative positions and extracts spans of Arg1 and Arg2. We use an efficient method to extract for integrating Arg1 and Arg2 together.

(3) **Explicit Sense Classifier** Determines the discourse function of the detected connectives.

For the explicit part and non-explicit part, there is:

(4) **Filter** Gets rid of obviously incorrect parts, such as the ones that have already been marked as explicit relationship, with the remainder as the input part of non-explicit.

The non-explicit part contains:

(5) **Non-explicit Argument Labeler** marks the location and range of Arg1 and Arg2 in the case that lacks of connective.

(6) **Non-explicit Sense Classifier** Determines the discourse relations according to the semantic context of Arg1 and Arg2.

3 System Components

Our system consists of six parts, and the general workflow refers to the shallow discourse parser based on the constituent parse tree (Chen et al., 2015). Feature extraction for training follows previous works (Kong et al., 2014; Lin et al., 2014; Pitler et al., 2009; Pitler and Nenkova, 2009).

We deduce each sentence into a constituent parse tree. Relative information is extracted from these constituent parse trees to train models and predict discourse relations.

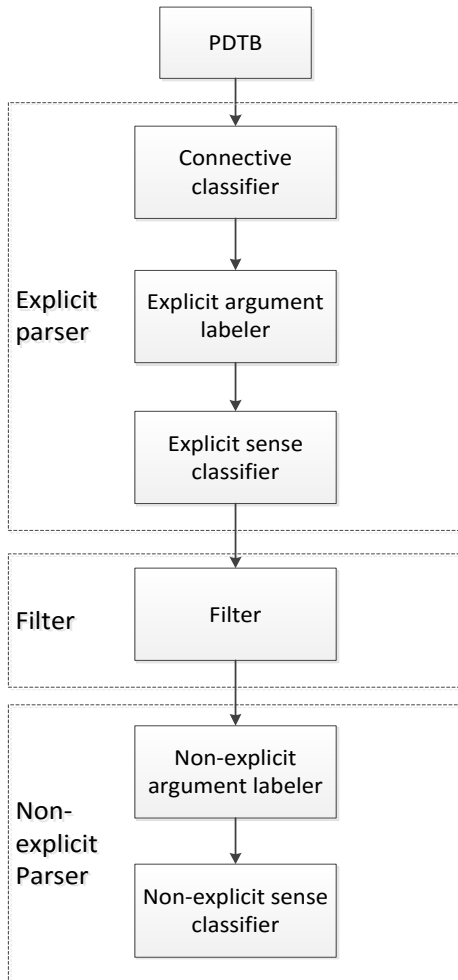


Figure 1: System overview

3.1 Explicit parser

3.1.1 Connective Classifier

In PDTB, there are 100 species of discourse connective, but not all conjunctions in the form of these 100 kinds of connective in the text are necessarily discourse relation. Thus, at first, we find out all connectives appearing in the text by scanning

each constituent parse tree, then use the connective classifier to determine whether each connective functions as discourse connective.

The features in connective classifier are as follows.

(1) **ConnPos** The category of the tree node which covers the whole connective.

(2) **PrevConn** The previous word of the connective and the connective itself.

(3) **PrevPos** The category of the previous word of the connective.

(4) **PrevPosConnPos** The category of previous word and category of the connective.

(5) **ConnNext** The connective itself and the next word of the connective.

(6) **NextPos** The category of the next word of the connective.

(7) **ConnPosNextPos** The category of the connective itself and category of the next word.

After extracting the mentioned feature for each connective, we annotate it as 1 or 0 according to whether this word in PDTB functions as discourse connective. (Jia et al., 2013; Zhao and Kit, 2008) showed maximum entropy classifier performed well in relative tasks, so we apply it to our classification problem*. According to official evaluation, F1 score of this part in our system is 0.9905 on the dev set and 0.9838 on the blind test set, comparing to 0.9514 and 0.9186, the best result of CoNLL-2015. The detailed results are shown in Table 1.

set	P	R	F
dev	0.9971	0.9840	0.9905
test	0.9967	0.9819	0.9892
blind	0.9856	0.9821	0.9838

Table 1: Official scores of connective classifier

From the comparison, we can learn that

(1) From the constituent parse tree we build, we can extract connective features precisely.

(2) We use a straightforward way build our classifier. Comparing to previous works, our features and model are much more intuitively, and finally get even better result.

(3) There are different ways to process text, and our work shows that using constituent parse tree is a proper method in this task or similar ones.

*MaxEnt classifier of OpenNLP, an open-source toolkit. See <http://opennlp.apache.org/>

3.1.2 Argument labeler

In this part, we use interval mapping based on constituent parse tree and the extracting method proposed by (Kong et al., 2014). When training, in constituent parse tree, we start with the node of connective, and ended with the root node. Along the path, left and right sibling of each node have become the candidate member of the Argument. Given that some part of the explicit discourse relation used previous sentence (PS) as Arg1, we use the a efficient method (Kong et al., 2014), which is to treat the sentence previous to the one contained discourse connective as the candidate of Arg1. Later, we compare these candidates with PDTB, and label them as Arg1 and Arg2 or null, according to their uses in PDTB, of which null means that the candidate doesn't have the function of Arg1 or Arg2. By this means, we obtain satisfying effect of argument labeling.

The features were as follows:

- (1) **ConStr** Prototype of connective in the text.
- (2) **ConLStr** Lowercase of connective.
- (3) **ConCat** Part of speech of connective.
- (4) **ConLSib** Left sibling number of connective.
- (5) **ConRSib** Right sibling number of connective.
- (6) **CandiCtx** Candidate's category, category of parent node, category of left sibling, and category of right sibling.
- (7) **ConCandiPath** Category of each node from the Candi to root node along the tree.
- (8) **ConCandiPosition** The relative position between Candi and connective (left or right).
- (9) **ConCandiPathLSib** Whether the left sibling number of the Candi is bigger than one.

3.1.3 Explicit Sense Classifier

In this part, we combine the feature of Lin's experiment with the feature of Pilter's, particularly as follows, (1) C prototype (2) C POS (3) prev+C (4) category of parent (5) category of left sibling (6) category of right sibling.

set	P	R	F
dev	0.4082	0.4219	0.4149
test	0.3226	0.3275	0.3251
blind	0.2527	0.2536	0.2531

Table 2: Official scores of explicit sense classifier

The detailed results are shown in Table 2. The results are also better than the best ones of CoNLL-2015, which were 0.3861 on the dev set and 0.2394 on the blind test set.

3.2 Filter

After identifying all explicit discourse relation connectives, and before non-explicit parser, we need to filter the training set. There are two cases for this filtering. (1) If one sentence, is labelled as Arg1 of some explicit discourse in previous step, then the related two sentences will not be considered by the following non-explicit parser. (2) In the original text, if two adjacent sentences are located between the last sentence of the previous paragraph and the first sentence of the next paragraph respectively, then these two sentences will not be considered, either.

3.3 Non-Explicit Parser

After explicit parser and filtering above, we take the rest part as input into non-explicit parser, for finding all the non-explicit discourse relations. In PDTB, there are three kinds of non-explicit discourse relations, which are Implicit, AltLex and EntRel. We notice that there is only 2.94% of AltLex. Besides, according to official evaluation criteria, we need to detect only 15 senses of the part of implicit. According to (Chen et al., 2015), we integrate EntRel together with implicit as a special sense for training and predicting.

3.3.1 Non-explicit Argument Labeler

In this part, we simply take the rest adjacent sentences which have been filtered as the argument span of non-explicit.

3.3.2 Non-explicit Sense Classifier

We perform sentence classification as mentioned above, practicing EntRel as a special sense of implicit, and ignored the senses which have few frequency of occurrences in PDTB. According to the previous works, the lost connective plays an important role in senses. Generally, connective appears at the beginning of the second sentence. According to this assumption, we use the following features.

- (1) **Arg1Last** The last word of Arg1.
- (2) **Arg1First** The first word of Arg1.
- (3) **Arg2Last** The last word of Arg2.
- (4) **Arg2First** The first word of Arg2.
- (5) **FirstS** The first word of Arg1 and Arg2.

- (6) **LastS** The last word of Arg1 and Arg2.
- (7) **Arg1First3** The first three words of Arg1.
- (8) **Arg1First3** The first three words of Arg2.
- (9) **Arg1Last3** The last three words of Arg1.

4 Results of Experiments

Our system is trained on the training set and evaluated on test set provided in the CoNLL-2016 Shared Task. We train our model of detecting connectives, extracting arguments of explicit part, predicting sense of connectives and predicting sense of non-explicit part, respectively.

The results of the official evaluation are shown in the Table 3, 4 and 5. From the result, we can learn that

- (1) The part of connective detection and classification achieve great performances.
- (2) The results of the sampled part are good, while there is still some gap between our system and the best one on the explicit and non-explicit part.

	P	R	F
Explicit Connective	0.9971	0.9840	0.9905
Extract Arg1	0.6128	0.5688	0.5900
Extract Arg2	0.7173	0.6658	0.6906
Extract Arg1&Arg2	0.4840	0.4493	0.4660
Parser	0.2778	0.3033	0.2900

Table 3: Official scores on dev set

	P	R	F
Explicit Connective	0.9967	0.9819	0.9892
Extract Arg1	0.5529	0.4988	0.5245
Extract Arg2	0.6674	0.6021	0.6331
Extract Arg1&Arg2	0.4033	0.3639	0.3826
Parser	0.2013	0.2233	0.2117

Table 4: Official scores on test set

	P	R	F
Explicit Connective	0.9856	0.9821	0.9838
Extract Arg1	0.5252	0.3501	0.4201
Extract Arg2	0.6675	0.4449	0.5339
Extract Arg1&Arg2	0.3615	0.2409	0.2891
Parser	0.1262	0.1894	0.1515

Table 5: Official scores on blind test set

5 Conclusion

In this paper, we present a complete discourse parser. Based on these previous works and through continuous improvement, our system has achieved good results. According to the official evaluation of CoNLL-2016 Shared Task closed track, our system gets 0.9905 in F1-measure on explicit connective classifier, and finally achieves 0.1515 in F1-measure on the official blind test.

References

- Changege Chen, Peilu Wang, and Hai Zhao. 2015. Shallow discourse parsing using constituent parsing tree. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 37–41, Beijing, China, July. Association for Computational Linguistics.
- Zhongye Jia, Peilu Wang, and Hai Zhao. 2013. Grammatical error correction as multiclass classification with single model. In *Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 74–81, Sofia, Bulgaria, August.
- Fang Kong, Hwee Tou Ng, and Guodong Zhou. 2014. A constituent-based approach to argument labeling with joint inference in discourse parsing. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 68–77, Doha, Qatar, October.
- Fang Kong, Sheng Li, and Guodong Zhou. 2015. The sonlp-dp system in the conll-2015 shared task. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 32–36, Beijing, China, July. Association for Computational Linguistics.
- Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, pages 151–184, April.
- Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 13–16, August.
- Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 683–691, August.

- Rashmi Prasad, Dinesh Nikhil, Alan Lee, Eleni Milt-sakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. 2008. The penn discourse treebank 2.0. *The International Conference on Language Resources and Evaluation*, May.
- Lianhui Qin, Zhisong Zhang, and Hai Zhao. 2016. Shallow discourse parsing using convolutional neural network. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning - Shared Task*, Berlin, Germany, August. Association for Computational Linguistics.
- Jianxiang Wang and Man Lan. 2015. A refined end-to-end discourse parser. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 17–24, Beijing, China, July. Association for Computational Linguistics.
- Rui Wang, Masao Utiyama, Isao Goto, Eiichiro Sumita, Hai Zhao, and Bao-Liang Lu. 2013. Converting continuous-space language models into n-gram language models for statistical machine translation. In *EMNLP*, pages 845–850.
- Peilu Wang, Zhongye Jia, and Hai Zhao. 2014a. Grammatical error detection and correction using a single maximum entropy model. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 74–82, Baltimore, Maryland, June. Association for Computational Linguistics.
- Peilu Wang, Yao Qian, Frank K Soong, Lei He, and Hai Zhao. 2014b. Learning distributed word representations for bidirectional lstm recurrent neural network. In *Proceedings of NAACL*.
- Rui Wang, Hai Zhao, Bao-Liang Lu, Masao Utiyama, and Eiichiro Sumita. 2014c. Neural network based bilingual language model growing for statistical machine translation. In *EMNLP*, pages 189–195.
- Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Atapol T. Rutherford, Bonnie Webber, Chuan Wang, and Hongmin Wang. 2016. The conll-2016 shared task on multilingual shallow discourse parsing. In *Proceedings of the Twentieth Conference on Computational Natural Language Learning: Shared Task*, Berlin, Germany.
- Hai Zhao and Chunyu Kit. 2008. Parsing syntactic and semantic dependencies with two single-stage maximum entropy models. *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 203–207, August.
- Hai Zhao, Wenliang Chen, and Chunyu Kit. 2009a. Semantic dependency parsing of nombank and propbank: An efficient integrated approach via a large-scale feature selection. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 30–39. Association for Computational Linguistics.
- Hai Zhao, Wenliang Chen, Chunyu Kit, and Guodong Zhou. 2009b. Multilingual dependency learning: A huge feature engineering method to semantic dependency parsing. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 55–60. Association for Computational Linguistics.
- Hai Zhao, Yan Song, Chunyu Kit, and Guodong Zhou. 2009c. Cross language dependency parsing using a bilingual lexicon. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 55–63. Association for Computational Linguistics.
- Hai Zhao, Xiaotian Zhang, and Chunyu Kit. 2013. Integrative semantic dependency parsing via efficient large-scale feature selection. *Journal of Artificial Intelligence Research*.
- Hai Zhao. 2009. Character-level dependencies in chinese: usefulness and learning. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 879–887. Association for Computational Linguistics.