UniTN End-to-End Discourse Parser for CoNLL 2016 Shared Task

Evgeny A. Stepanov and **Giuseppe Riccardi**
Signals and Interactive Systems Lab
Department of Information Engineering and Computer Science
University of Trento, Trento, TN, Italy
{evgeny.stepanov,giuseppe.riccardi}@unitn.it

Abstract

Penn Discourse Treebank style discourse parsing is a composite task of detecting explicit and non-explicit discourse relations, their connective and argument spans, and assigning a sense to these relations. Due to the composite nature of the task, the end-to-end performance is greatly affected by the error propagation. This paper describes the end-to-end discourse parser for English submitted to the CoNLL 2016 Shared Task on Shallow Discourse Parsing with the main focus of the parser being on argument spans and the reduction of global error through model selection. In the end-to-end closed-track evaluation the parser achieves F-measure of 0.2510 outperforming the best system of the previous year.

1 Introduction

Discourse parsing is a Natural Language Processing (NLP) task with the potential utility for many other Natural Language Processing tasks (Webber et al., 2011). However, as was illustrated by the CoNLL 2015 Shared Task on Shallow Discourse Parsing (Xue et al., 2015), the task of Penn Discourse Treebank (PDTB) (Prasad et al., 2008) style discourse parsing is very challenging as the best system achieved the end-to-end parsing performance of $F_1 = 0.24$. The main reason for the low performance is the composite nature of the task and the error propagation through the long pipeline.

In PDTB discourse relations are binary: a discourse connective and its two arguments. The arguments are defined syntactically such that *Argument 2* is syntactically attached to the connective, and *Argument 1* is the other argument. A discourse relation is assigned a particular sense from the predefined sense hierarchy. Discourse connective, a member of the closed class, signals the presence of an *explicit* relation. Besides explicit discourse relations there are non-explicit relations: *implicit* relations where a connective is implied and can be inserted, *alternative lexicalizations* (AltLex) where a connective cannot be inserted and a relation is signaled by a phrase not in the list of discourse connectives, and *entity relations* (EntRel) where two arguments share the same entity.

Such definition of discourse relations naturally suggests at least two pipelines for the parsing: for explicit and non-explicit relations. Moreover, since in PDTB non-explicit relations are annotated only in the absence of explicit relations, explicit relation parsing pipeline precedes the non-explicit one. While detection of discourse connectives is only required for the explicit relations, for both relation types parsing requires identification of argument spans and relation senses. Consequently, PDTB-style discourse parsing is partitioned into several sub-tasks: (1) explicit discourse connective detection, (2) argument span extraction (with labeling for *Argument 1* and *2*), and (3) sense classification. The tasks are often conditioned on the type of a relation (explicit or non-explicit) and argument positions (intra- or inter-sentential).

In this paper we describe the end-to-end discourse parser submitted to CoNLL 2016 Shared Task on Shallow Discourse Parsing (Xue et al., 2016). The parser makes use of token-level sequence labeling with Conditional Random Fields (Lafferty et al., 2001) for the identification of connective and argument spans; and classification for the identification of relation senses and argument positions. The main focus of the parser is on argument spans. For the end-to-end parsing task the models are selected with respect to the global parsing score.

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 85–91,
Berlin, Germany, August 7-12, 2016.

The overall parser architecture is described in Section 1. The token-level features used for sequence labeling and argument and relation-level features used for sense classification are described in Section 3. The individual discourse parsing subtasks are described in Section 4. Section 5 describes the official CoNLL 2016 Shared Task evaluation results, and in Section 6 we compare the system to the best systems of the preceding shared task on discourse parsing (Xue et al., 2015). Section 7 provides concluding remarks.

2 System Architecture

The discourse parser submitted for the CoNLL 2016 Shared Task is the modified version of the parser developed by (Stepanov et al., 2015) for the shared task of 2015. The system is an extension of the explicit relation parser described in (Stepanov and Riccardi, 2013; Stepanov and Riccardi, 2014). The overall architecture of the parser is depicted in Figure 1. The approach implements discourse parsing as a pipeline of several tasks such that connective and argument span decisions are cast as sequence labeling and sense decisions as classification.

The discourse parsing pipelines starts with the identification of discourse connectives and their spans (*Discourse Connective Detection* (DCD)), and is followed by *Connective Sense Classification* (CSC) and *Argument Position Classification* (APC) steps. While CSC assigns sense to explicit discourse relations, APC classifies them as intra- and inter-sentential (*Same Sentence* (SS) and *Previous Sentence* (PS) *Argument 1*). Both tasks operate using the connective span tokens only.

With respect to the decision of the *Argument Position Classification* the pipeline is split into explicit and non-explicit tasks. For the explicit relations, specific *Argument Span Extraction* (ASE) models are applied for each of the arguments with respect to their begin intra- or inter-sentential. Since *Argument 2* is syntactically attached to the discourse connective, its identification is easier. Thus, for the intra-sentential (SS) relations, models are applied in a cascade such that the output of *Argument 2* span extraction in the input for *Argument 1* span extraction. For the inter-sentential (PS) relations, on the other hand, a sentence containing the connective is selected as *Argument 2*, and the sentence immediately preceding it as a candidate for *Argument 1*.

For non-explicit discourse relations, a set of candidate argument pairs is constructed using adjacent sentence pairs within a paragraph and removing all the sentence pair already identified as inter-sentential explicit relations. Each of these argument pairs is assigned a sense using *Non-Explicit Relation Sense Classification* (NE-RSC) models and their argument spans are extracted using *Non-Explicit Argument Span Extraction* step.

In the discourse parser, the *Non-Explicit Relation Sense Classification*, *Connective Sense Classification*, and *Argument Position Classification* tasks are cast as supervised classification using AdaBoost algorithm (Freund and Schapire, 1997) implemented in *icsiboost* (Favre et al., 2007). The span extraction tasks (*Discourse Connective Detection* and explicit and non-explicit *Argument Span Extraction*), on the other hand, are cast as token-level sequence labeling with CRFs (Lafferty et al., 2001) using CRF++ (Kudo, 2013). Besides training the CRF models for ASE, for inter-sentential *Argument 1* span and both non-explicit argument spans, we also make use of the 'heuristics': taking an argument sentence as a whole and removing leading and trailing punctuation (Lin et al., 2014; Stepanov et al., 2015). In the next section we describe the features used for the tasks.

3 Features

The PDTB corpus distributed to the shared task participants contains raw text and syntactic constituency and dependency parses. Besides the token and part-of-speech tags, these resources are used to extract and generate both token-level and argument/relation-level features. Additionally, for argument/relation-level features for *Non-Explicit Relation Sense Classification* we make use of Brown Clusters (Turian et al., 2010), MPQA subjectivity lexicon (Wilson et al., 2005) and VerbNet (Kipper et al., 2008). The feature sets for each task are selected using greedy hill climbing approach, also considering the amount of contribution of each individual feature.

3.1 Token-level Features

All the discourse parsing sub-tasks (both classification and sequence labeling) except *Non-Explicit Relation Sense Classification* make use of token-level features. However, the feature sets for each task are different. Table 1 gives an overview of feature sets per task. Besides tokens and POS-

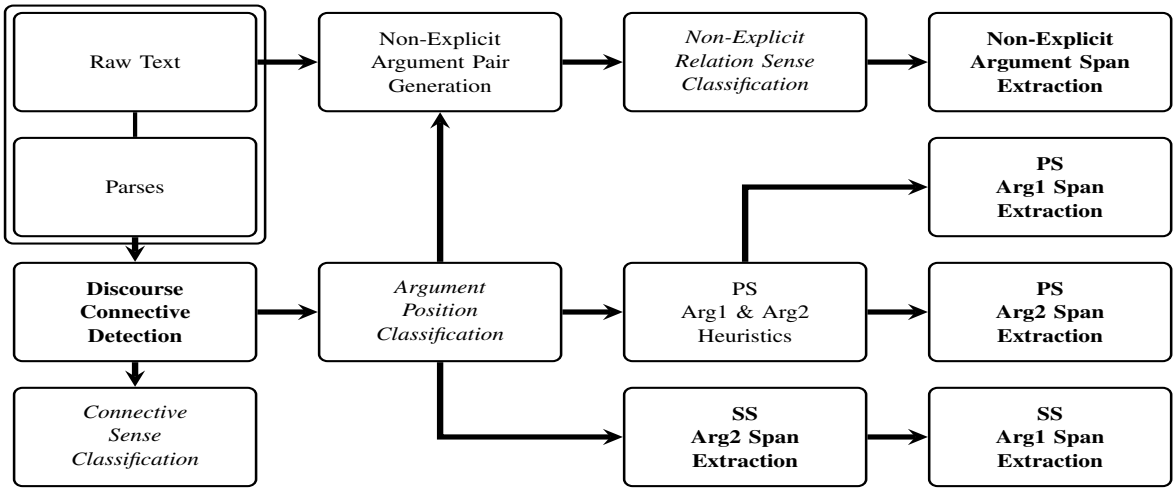

Figure 1: Discourse parsing architecture: the sequence labeling modules are in **bold** and the classification modules are in *italics*.

tags, the rest of the features are described below.

Chunk-tag is the syntactic chunk prefixed with the information whether a token is at the beginning (B-), inside (I-) or outside (O) of the constituent (i.e. IOB format) (e.g. 'B-NP' indicates that a token is at the beginning of Noun Phrase chunk). The information is extracted from constituency parse trees using chunklink script (Buchholz, 2000).

IOB-chain is the path string of the syntactic tree nodes from the root node to the token, similar to Chunk-tag, it is prefixed with the IOB information. For example, the IOB-chain 'I-S/B-VP' indicates that a token is the first word of the verb phrase (B-VP) of the main clause (I-S).The feature is also extracted using the chunklink script (Buchholz, 2000).

Dependency chain (Stepanov et al., 2015) is a feature inspired by *IOB-chain* and is the path string of the functions of the parents of a token, starting from the root of a dependency parse.

VerbNet Class (Kipper et al., 2008) is a feature intended to capture attributions. The feature requires lemmas, which were extracted using TreeTagger (Schmid, 1995).

Connective Label and *Argument 2 Label* are the output labels of the *Discourse Connective Detection* and *Argument 2 Span Extraction* models respectively.

Using templates of CRF++ the token-level features are enriched with ngrams (2 & 3-grams) in the window of ± 2 tokens, such that for each token there are 12 features per feature type: 5 unigrams, 4 bigrams and 3 trigrams. All features are conditioned on the output label independently of each other. Additionally, CRFs consider the previous token's output label as a feature.

3.2 Argument and Relation-level Features

In this section we describe the features used for *Non-Explicit Relation Sense Classification*. Previous work on the task makes use of a wide range of features; however, due to the low state-of-the-art on the task, we focused on the features obtainable from the provided resources: sentiment polarities from MPQA lexicon (Wilson et al., 2005), Brown Clusters (Turian et al., 2010), and VerbNet (Kipper et al., 2008). Similar to VerbNet Class feature, described above, lemmas from TreeTagger (Schmid, 1995) are used to compute the polarity features.

There are four features generated for *Polarity*: (1-2) Individual argument polarities computed from token-level polarities as a difference of counts of positive and negative polarity words. The feature is assigned either 'negative' or 'positive' value with respect to the difference. (3) The concatenation of the argument polarity values (e.g. *negative-positive*). (4) The boolean feature indicating whether the argument polarities match.

The Brown Cluster and VerbNet features are extracted only for specific tokens. Starting from the dependency parse trees of the arguments we extract the main verb (root), subject (including passive), direct and indirect objects for each of them. Since for extracting VerbNet features we make use of lemmas, the lemmas themselves are considered for classification as well. Similar to polarity, the *VerbNet* features (4) are main-verbs' classes of the

Feature	DCD	CSC	APC	ASE: SS		ASE: PS		NE-ASE	
				A1	*A2*	*A1*	*A2*	*A1*	*A2*
Token	Y	Y	Y	Y	Y	Y	Y	Y	Y
POS-tag	Y		Y		Y	Y	Y	Y	Y
Chunk-tag	Y								
IOB-chain	Y		Y	Y	Y	Y	Y	Y	Y
Dependency chain					Y				
VerbNet class					Y				
Connective Label				Y	Y		Y		
Argument 2 Label				Y					

Table 1: Token-level features for classification and sequence labeling tasks: Discourse Connective Detection (DCD), Connective Sense Classification (CSC), Argument Position Classification (APC), and Argument Span Extraction (ASE) of intra- (SS) and inter-sentential (PS) explicit and non-explicit (NE) relations.

arguments, their concatenation, and a boolean feature indicating their match.

The *Brown Cluster* and *Lemma* features are main-verbs' brown clusters and lemmas, their concatenation and boolean features for matches (4). Unlike VerbNet, these features are also generated for a Cartesian product for the arguments' subject, direct and indirect objects. Consequently, there are 4 features for verbs and 24 for other dependency roles (3 + 3 + 9 + 9) per feature type.

4 Individual Modules

In this section we provide implementation details for the individual components of the discourse parser. We first address explicit and then non-explicit relations.

4.1 Explicit Discourse Relations

The explicit relation pipeline consists of *Discourse Connective Detection*, *Connective Sense Classification*, *Argument Position Classification* and *Argument Span Extraction* tasks.

4.1.1 Discourse Connective Detection

Since *Discourse Connective Detection* is the first step in discourse parsing, the performance of the task is critical. The task is cast as sequence labeling with CRFs. The performance of the models is tuned by feature ablation to yield a model that achieves F_1 of 0.9332 on the development set. The best model is trained on cased tokens, POS-tags, Chunk-tag and IOB-chain features.

4.1.2 Connective Sense Classification

Following (Stepanov et al., 2015) the *Connective Sense Classification* step assigns a sense to a connective considering only cased tokens. The classification is performed directly into 14 explicit relation senses.

4.1.3 Argument Position Classification

Due to the fact that explicit discourse connectives have a strong preference on the positions of their arguments, depending on whether they appear at the beginning or in the middle of a sentence (Stepanov and Riccardi, 2013), the task is easy. The features used for the task are cased tokens, POS-tags and IOB-chains. Case of the tokens carries position information. The accuracy on the development set without error propagation is 0.9868.

4.1.4 Argument Span Extraction

Argument Span Extraction is the main focus of the development for the submission. We train CRF model for each of the arguments of the intra- and inter-sentential relations considering a single sentence as a candidate (i.e. all multi-sentence relations are missed). As a candidate for the inter-sentential *Argument 1* we consider only immediately preceding sentence (effectively missing all non-adjacent *Argument 1* relations).

Since *Argument 2* models make use of connective span labels as a feature, and intra-sentential *Argument 1* model makes use of both connective and *Argument 2* labels; these models are trained using reference annotation spans. For the *Argument Span Extraction* of inter-sentential *Argument 1*, additional to the training of the CRF models we also make use of the heuristic, that takes the sentence as a whole and removes leading and trailing punctuation.

There are 4 CRF models for the task with the additional heuristic for the inter-sentential *Argument 1*. The feature sets for each of the models are selected such that they maximize the F-measure of both arguments together.

The CRF model for the inter-sentential *Argument 1* yields higher performance than the heuristic. However, the submitted system exploits the heuristic, since the difference between the two for the both argument spans is not large (0.4981 vs. 0.4936 for the heuristic).

4.2 Non-Explicit Discourse Relations

The non-explicit relation parsing pipeline consists of *Relation Sense Classification* (NE-RSC) and *Argument Span Extraction* (NE-ASE) tasks. Even though, NE-ASE is applied after NE-RSC with the idea of exploiting classification confidences for filtering out the candidate relations, the two tasks are fairly independent.

4.2.1 Non-Explicit Relation Sense Classification

The set of features for the task is described in Section 3. It is the only task that makes use of the argument and relation level features. Due to the low state-of-the-art on the task, the focus is on the development of the models that maximize the performance of the majority senses – *EntRel* and *Expansion.Conjunction*. The flat classification mode is considered as it yields higher performance for these senses (e.g. for EntRel the classification into 4 top-level senses + EntRel yields F_1 of $\approx$ 0.30, while flat classification into 14 full senses + EntRel F_1 of 0.44).

4.2.2 Non-Explicit Argument Span Extraction

The task is implemented similar to the *Argument Span Extraction* of the inter-sentential *Argument 1*, and considers the same feature set (cased token, POS-tag, and IOB-chain). Similarly, we experiment with the span extraction heuristic by only removing leading and trailing punctuation.

Unlike explicit relations, the CRF models for the non-explicit argument span extraction perform significantly better than the heuristics. However, due to the error propagation from the Relation Sense Classification task, the heuristics yield the higher F_1-measure for the end-to-end parsing of non-explicit relations. Thus, the submitted system contains purely heuristic *Non-Explicit Argument Span Extraction*.

5 Official Evaluation Results

The official end-to-end parsing evaluation of the CoNLL 2016 Shared Task on Shallow Discourse Parsing carried on TIRA platform (Potthast et al., 2014) is on a per-discourse relation basis. A relation is considered to be predicted correctly only in case the parser correctly predicts (1) discourse connective head, (2) exact spans and labels of both arguments, and (3) sense of a relation. The official evaluation is reported for the PDTB development and test sets (sections 22 and 23, respectively) and a blind test set.

The reported evaluation metrics are (1) explicit discourse connective, (2-4) *Argument 1* and *Argument 2* spans individually and together, and the sense of a relation. The reported micro-F_1 measure of the sense classification is equivalent to the end-to-end parsing performance as it considers the error propagation from the upstream tasks. The metrics are reported for explicit and non-explicit relations individually and jointly. The performance of the submitted system on all the metrics is reported in Table 2. On the closed-track evaluation, the system achieves end-to-end parsing F_1 of 0.3246, 0.2789 and 0.2510 on the development, test and blind test sets respectively.

6 Comparison to CoNLL 2015 Systems

The current shared task is the second edition of the CoNLL Shared Task on Shallow Discourse Parsing. Thus, it makes sense to compare the performances of the submission to the systems of the first edition (i.e. the winner (Wang and Lan, 2015) and (Stepanov et al., 2015), which is taken as the baseline). Since the submitted system is an extension of (Stepanov et al., 2015), the main focus of the comparison is on the changes and their effects on the performance.

We first compare the system performance to the last year's systems on the end-to-end parsing score on the blind test set (see Table 3). The current submission outperforms the baseline (Stepanov et al., 2015) as well as the best system (ECNU) (Wang and Lan, 2015). The recall of the 2015 winner is slightly higher (0.2407 vs. 0.2432 for ECNU); however, the difference is well compensated by the higher precision (0.2622 vs. 0.2369 for ECNU).

Task	All Relations			Explicit			Non-Explicit		
	Dev	**Test**	**Blind**	**Dev**	**Test**	**Blind**	**Dev**	**Test**	**Blind**
Connective	0.9332	0.9243	0.8856	0.9332	0.9243	0.8856	–	–	–
Arg 1	0.6417	0.5890	0.5991	0.5566	0.4964	0.5028	0.6951	0.6558	0.6683
Arg 2	0.7664	0.7188	0.7586	0.7907	0.7651	0.7205	0.7451	0.6778	0.7911
Arg 1+2	0.5471	0.4844	0.5060	0.4936	0.4456	0.4184	0.5940	0.5180	0.5805
Parser	0.3246	0.2780	0.2510	0.4589	0.3960	0.3174	0.2089	0.1756	0.1946

Table 2: Task-level and end-to-end F_1-measures of the discourse parser on the development, test, and blind test sets for explicit and non-explicit relations individually and jointly for all relations. The task-level performances are reported with the error propagation. Thus, the *sense classification* performances are equivalent to the end-to-end parser performances.

System	P	R	F
our system	**0.2622**	0.2407	**0.2510**
ECNU	0.2369	**0.2432**	0.2400
(Stepanov et al., 2015)	0.2094	0.2283	0.2184

Table 3: Precision (**P**), recall (**R**) and F_1 (**F**) of the end-to-end discourse parsing on the blind test set for the best CoNLL 2015 Shared Task systems and the current submission.

System	Dev	Test	Blind
Arg 1+2 Span Extraction			
our system	**0.5940**	**0.5180**	**0.5805**
(Stepanov et al., 2015)	0.4000	0.3730	0.3831
Non-Explicit Parsing			
our system	**0.2089**	**0.1756**	**0.1946**
(Stepanov et al., 2015)	0.1577	0.1330	0.1577

Table 4: F_1 for the non-explicit argument extraction and parsing.

The major change from (Stepanov et al., 2015) is the elimination of the *Non-Explicit Relation Detection* step. The step classified non-explicit relation candidates into relations and non-relations. However, the ratio of non-related adjacent sentence pairs in the PDTB is very low (circa 1%). Consequently, the step was penalizing the performance on non-explicit relations. As it can be observed from Table 4, there is a major improvement in performance for non-explicit argument spans.

The other changes are in the feature sets of *Connective Detection* and the *Argument Span Extraction* of the explicit intra-sentential *Argument 2*. For the former we improved the performance on the development set, but the performance on the test and blind test sets dropped (see Table 5). For the latter, we introduced a new feature – VerbNet (Kipper et al., 2008) classes – intended to capture the attribution spans. From the results it appears that the feature is useful, as they are better than

System	Dev	Test	Blind
Discourse Connective Detection			
our system	**0.9332**	0.9243	0.8856
(Stepanov et al., 2015)	0.9219	**0.9271**	**0.8992**
Explicit SS Arg 2			
our system	**0.7907**	**0.7651**	**0.7205**
(Stepanov et al., 2015)	0.7748	0.7616	0.7068

Table 5: F_1 for the *Discourse Connective Detection* and explicit intra-sentential *Argument 2* span extraction.

the results of (Stepanov et al., 2015) despite the lower connective detection performance.

7 Conclusions

In this paper we have presented the parser submitted to CoNLL 2016 Shared Task on Shallow Discourse Parsing. The parser is a modified version of the system of (Stepanov et al., 2015). We have described the discourse parsing architecture and models for each of the sub-tasks. The distinct feature of the approach is casting the span extraction tasks are token-level sequence labeling with Conditional Random Fields. The focus of the development for the shared task was on *Argument Span Extraction* and its optimization for the end-to-end parsing score on the development set. The main change made to the baseline version of the system is the elimination of non-explicit relation detection step, which boosted the overall performance of the system to outperform the CoNLL 2015 Shared Task winner.

Acknowledgments

The research leading to these results has received funding from the European Union – Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 610916 – SENSEI.

References

Sabine Buchholz. 2000. chunklink.pl. http://ilk.uvt.nl/software/.

Benoit Favre, Dilek Hakkani-Tür, and Sebastien Cuendet. 2007. Icsiboost. https://github.com/benob/icsiboost/.

Yoav Freund and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A large-scale classification of english verbs. *Language Resources and Evaluation Journal*, 42(1):21–40.

Taku Kudo. 2013. CRF++. http://taku910.github.io/crfpp/.

John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of 18th International Conference on Machine Learning*, pages 282–289.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, 20:151 – 184.

Martin Potthast, Tim Gollub, Francisco Rangel, Paolo Rosso, Efstathios Stamatatos, and Benno Stein. 2014. Improving the Reproducibility of PAN's Shared Tasks: Plagiarism Detection, Author Identification, and Author Profiling. In *Information Access Evaluation meets Multilinguality, Multimodality, and Visualization. 5th International Conference of the CLEF Initiative (CLEF 14)*, pages 268–299, Berlin Heidelberg New York.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse Treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to german. In *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland.

Evgeny A. Stepanov and Giuseppe Riccardi. 2013. Comparative evaluation of argument extraction algorithms in discourse relation parsing. In *The 13th International Conference on Parsing Technologies (IWPT 2013)*, pages 36–44, Nara, Japan, November. ACL.

Evgeny A. Stepanov and Giuseppe Riccardi. 2014. Towards cross-domain PDTB-style discourse parsing. In *EACL Workshops - The Fifth International Workshop on Health Text Mining and Information Analysis (Louhi 2014)*, pages 30–37, Gothenburg, Sweden, April. ACL.

Evgeny A. Stepanov, Giuseppe Riccardi, and Ali Orkan Bayer. 2015. The UniTN discourse parser in CoNLL 2015 shared task: Token-level sequence labeling with argument-specific models. In *Proceedings of the SIGNLL Conference on Computational Natural Language Learning (CoNLL) - Shared Task*, pages 25–31, Beijing, China, July. ACL.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semisupervised learning. In *In ACL*, pages 384–394.

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*.

Bonnie L. Webber, Markus Egg, and Valia Kordoni. 2011. Discourse structure and language technology. *Natural Language Engineering*, pages 1–54.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP)*, Vancouver, B.C., Canada, October.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Rashmi Prasad, Christopher Bryant, and Attapol Rutherford. 2015. The CoNLL-2015 Shared Task on Shallow Discourse Parsing. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning: Shared Task*.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Bonnie Webber, Attapol Rutherford, 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, August. Association for Computational Linguistics.