The UniTN Discourse Parser in CoNLL 2015 Shared Task: Token-level Sequence Labeling with Argument-specific Models

Evgeny A. Stepanov Giuseppe Riccardi Ali Orkan Bayer
Signals and Interactive Systems Lab
Department of Information Engineering and Computer Science
University of Trento, Trento, TN, Italy
{stepanov,riccardi,bayer}@disi.unitn.it

Abstract

Penn Discourse Treebank style discourse parsing is a composite task of identifying discourse relations (explicit or non-explicit), their connective and argument spans, and assigning a sense to these relations from the hierarchy of senses. In this paper we describe University of Trento parser submitted to CoNLL 2015 Shared Task on Shallow Discourse Parsing. The span detection tasks for explicit relations are cast as token-level sequence labeling. The argument span decisions are conditioned on relations' being intra- or inter-sentential. Non-explicit relation detection and sense assignment tasks are cast as classification. In the end-to-end closed-track evaluation, the parser ranked second with a global F-measure of 0.2184

1 Introduction

Discourse parsing is a challenging Natural Language Processing (NLP) task that has utility for many other NLP tasks such as summarization, opinion mining, etc. (Webber et al., 2011). With the release of Penn Discourse Treebank (PDTB) (Prasad et al., 2008), the researchers have developed discourse parsers for all (e.g. (Lin et al., 2014) or some (e.g. (Ghosh et al., 2011)) discourse relation types in the PDTB definition, or addressed particular discourse parsing subtasks (Pitler and Nenkova, 2009).

PDTB adopts non-hierarchical binary view on discourse relations: a discourse connective and its two arguments – *Argument 1* and *Argument 2*, which is syntactically attached to the connective. And, a relation is assigned particular sense from the sense hierarchy. It was identified that parsing *Explicit* discourse relations, that are signaled by a presence of a discourse connective (a closed class), is much easier task than detection and classification of *Implicit* discourse relations, where a discourse connective is implied, rather than lexically realized. Since Explicit and Implicit discourse relations in a document do not differ much in relative frequency, the low performance on one of the relation types limits the utility of discourse parsing for downstream applications.

In this paper we describe the University of Trento discourse parser for both explicit and non-explicit – implicit, alternatively lexicalized (AltLex), and entity (EntRel) relations – that was submitted to the CoNLL 2015 Shared Task on Shallow Discourse Parsing (Xue et al., 2015) and ranked 2nd. The parser makes use of token-level sequence labeling with Conditional Random Fields (Lafferty et al., 2001) for identification of connective and argument spans; and classification for identification of relation senses and argument configurations.

The parser architecture is described in Section 2. The features and individual model details are described in Sections 3 and 4, respectively. In Section 5 we describe official evaluation results. Section 6 discusses the lessons learned from the shared task and provides concluding remarks.

2 System Architecture

The discourse parser submitted for the CoNLL 2015 Shared Task is the extension of the parser described in (Stepanov and Riccardi, 2013; Stepanov and Riccardi, 2014). The overall architecture of the parser is depicted in Figure 1. The approach structures discourse parsing into a pipeline of several subtasks, mimicking the Penn Discourse Treebank (PDTB) (Prasad et al., 2008) annotation procedure as in (Lin et al., 2014).

The first step is *Discourse Connective Detection* (DCD) that identifies explicit discourse connectives and their spans. Then *Connective Sense Classification* (CSC) is used to classify these con-

Proceedings of the Nineteenth Conference on Computational Natural Language Learning: Shared Task, pages 25–31,
Beijing, China, July 26-31, 2015.

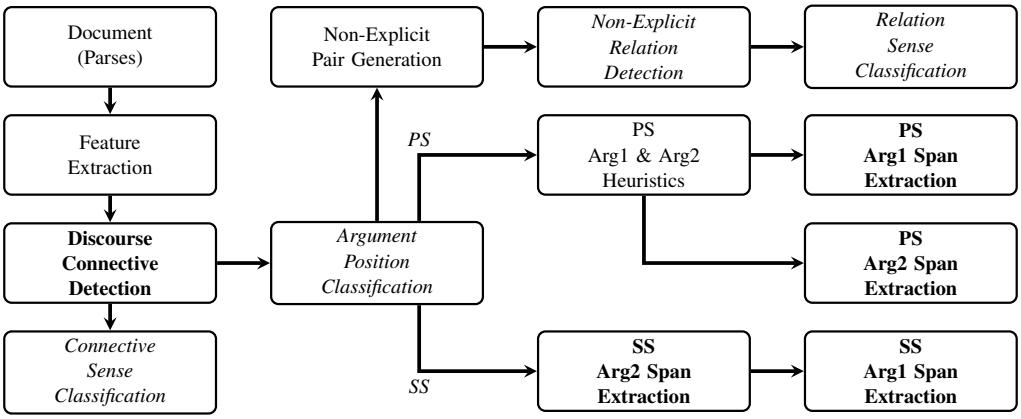

Figure 1: Discourse parser architecture. CRF modules are in **bold**; classification modules are in *italic*.

nectives into the PDTB hierarchy of senses; and *Argument Position Classification* (APC) to classify the connectives as requiring their *Argument 1* in the previous (PS) or the same sentence as *Argument 2* (i.e. classify relations as inter- and intra-sentential). With respect to the decision of the step an *Argument Span Extraction* (ASE) model is applied to label the spans of both arguments.

Separate *Argument Span Extraction* models are trained for each of the arguments of intra- and inter-sentential explicit discourse relations. Identification of *Argument 2* is much easier, since it is the argument syntactically attached to the discourse connective. 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, a sentence containing the connective is selected as *Argument 2*, and the sentence immediately preceding it as a candidate for *Argument 1*. Even though in 9% of all inter-sentential relations *Argument 1* is located in non-adjacent previous sentence (Prasad et al., 2008), this heuristic is widely used (Lin et al., 2014; Stepanov and Riccardi, 2013), and is known as Previous Sentence Heuristic.

In PDTB, the Non-Explicit discourse relations – Implicit, AltLex, and EntRel – are annotated for pairs of adjacent sentences except the pairs that were already annotated as explicit discourse relations (Prasad et al., 2007). Thus, in the *Non-Explicit Pair Generation* (NPG) step a list of adjacent sentence pairs is generated omitting the inter-sentential explicit relations identified in the APC step. In the *Non-Explicit Relation Detection* (NRD) step the candidate pairs are classified as holding a relation or not. The pairs identified as a relation are then classified into relation senses in the *Relation Sense Classification* (RSC) step.

Since the goal of *Discourse Connective Detection* and *Argument Span Extraction* tasks is to label the *spans* of a connective and its arguments, they are cast as token-level sequence labeling with CRFs using *CRF++* (Kudo, 2013). The *Non-Explicit Relation Detection* and *Sense* and *Argument Position* classification tasks are cast as supervised classification using AdaBoost algorithm (Freund and Schapire, 1997) implemented in *icsiboost* (Favre et al., 2007). In Section 3 we describe the features used for token-level sequence labeling and classification tasks; and in Section 4 models for each of the subtasks in more detail.

3 Features

Besides tokens, the PDTB corpus distributed to the participants contains Part-of-Speech tags, constituency and dependency parses. These resources are used to extract and generate both token-level and argument/relation-level features. Additionally, for argument/relation-level features Brown Clusters (Turian et al., 2010) are used.

3.1 Token-level Features

Discourse Connective Detection and *Argument Span Extraction* tasks of discourse parsing are cast as token-level sequence labeling with CRFs. The list of features used for the models is given in Table 1. Besides tokens and POS-tags, the rest of the features is 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

Feature	DCD	ASE: SS		ASE: PS	
		A1	*A2*	*A1*	*A2*
Token	Y	Y	Y	Y	Y
POS-tag	Y		Y	Y	Y
Chunk-tag	Y				
IOB-chain	Y	Y	Y	Y	Y
Dependency chain	Y		Y		
Connective Head	Y				
Connective Label		Y	Y		Y
Argument 2 Label		Y			

Table 1: Token-level features for Discourse Connective Detection (DCD) and Argument Span Extraction (ASE) for intra-sentential (SS) and inter-sentential (PS) explicit discourse relations.

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 is a feature inspired by *IOB-chain* and is the path string of the functions of the parents of a token, starting from root of a dependency parse. For example, the dependency chain 'root/nsubj/det' indicates that a token is a determiner of the subject of a sentence.

Connective Head is a binary feature that indicates whether a token is in the list of 100 PDTB discourse connectives. For example, all 'and' tokens will have this feature value '1'.

Connective Label and *Argument 2 Label* are the output labels of the *Discourse Connective Detection* and *Argument 2 Span Extraction* models respectively. The outputs are the IOB-tagged strings 'CONN' and 'ARG2'. Using these labels as features for Argument Span Extraction is useful for constraining the search space, since the *Connective*, *Argument 1* and *Argument 2* spans are not supposed to overlap.

Besides the features mentioned above, we have experimented with other token-level features: (1) morphological: lemma and inflection; (2) dependency: main verb of a sentence (i.e. root of the dependency parse) as a string and binary feature; (3) Connective Head as string. Even though previous work on discourse parsing (e.g. (Ghosh et al., 2011; Stepanov and Riccardi, 2013) found these features useful in token-level sequence labeling approach to *Argument Span Extraction* using gold parse trees, they were excluded from the submitted models since in greedy hill climbing their contributions were negative.

Using templates of CRF++ the token-level features are enriched with ngrams (2 & 3-grams) in the window of ± 2 tokens. That is, 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 & Relation-level Features

In this section we describe features used for detecting non-explicit discourse relations and their sense classification. Since in these tasks the unit of classification is a relation rather than token, these features are extracted per argument of a relation and a relation as a whole.

Previous work on the topic makes use of wide range of features ranging from first and last tokens of arguments to a Cartesian product of all tokens in both arguments, which leads to a very sparse feature set. To reduce the sparseness in (Rutherford and Xue, 2014) the authors map the tokens to Brown Clusters (Turian et al., 2010) and improve the classification into top-level senses.

Inspired by the previous research, we have experimented with the following features that are extracted from both arguments:

1. Bag-of-Words;
2. Bag-of-Words prefixed with the argument ID (Arg1 or Arg2);
3. Cartesian product of all the tokens from both arguments;
4. Set of unique pairs from Cartesian product of Brown Clusters of all the tokens from both arguments (inspired by (Rutherford and Xue, 2014));
5. First, last, and first 3 words of each argument (from (Pitler et al., 2009; Rutherford and Xue, 2014));
6. Predicate, subject (both passive and active), direct and indirect objects, extracted from dependency parses (8 features);

7. Ternary features for pairs from 6 to indicate matches (1, 0) or NULL, if one of the arguments misses the feature (extension of 'similar subjects or main predicates' feature of (Rutherford and Xue, 2014)) (16 features);
8. Cartesian product of Brown Clusters of 6 (16 features);

These features are used for *Non-Explicit Discourse Relation Detection* and *Sense Classification* tasks, which are described in the next section.

4 Discourse Parsing Components

In this section we describe individual discourse parsing subtasks discussing features and models.

4.1 Discourse Connective Detection

Discourse Connective Detection is the first step in discourse parsing. The CRF model makes use of all the features in Table 3 (except Connective Label – its own output – and Argument 2 Label – the output of downstream component). Using just cased token features (i.e. 1, 2, 3-grams in the window of ± 2 tokens already has F-measure above 0.85. Adding other features gradually increases the performance on the development set to 0.9379. Other than the token itself, the feature that contributes the most to the performance is IOB-chain.

4.2 Connective Sense Classification

Connective Sense Classification takes the output of *Discourse Connective Detection* and classifies identified connectives into the hierarchy of PDTB senses. We have experimented with two approaches: (1) flat – directly classifying into full spectrum of senses including class, type and subtype (Prasad et al., 2008); and (2) hierarchical – first classifying into 4 top level senses (Comparison, Contingency, Expansion and Temporal) and then into the rest of the levels. For the purposes of the Shared Task partial senses (e.g. just class) were disallowed; thus, for the flat classification, instances having partial senses were removed from both training and development sets.

The flat classification into 14 senses using just cased token strings as bag-of-words yields the best performance and has accuracy of 0.8968 on the filtered development set using gold connective spans. The 4-way classification into top-level senses on a full development set using just connective tokens has accuracy of 0.9426. Adding POS-tags increases accuracy to 0.9456. Due to the error propagation, going to the second level of the hierarchy drops the performance slightly below the flat classification. None of the other features listed in Table 1 has a positive effect on classification. Adding argument spans lowered the performance as well.

4.3 Argument Position Classification

Argument Position Classification is an easy task, since explicit discourse connectives have a strong preference on the positions of its arguments, depending on whether they appear at the beginning or in the middle of a sentence. In the literature the task was reported as having a very high baseline (e.g. (Stepanov and Riccardi, 2013), 95% for whole PDTB). The features used for classification are cased connective token string (case here carries the information about connective's position in the sentence), POS-tags and IOB-chains. The accuracy on the development set given gold connective spans is 0.9868.

4.4 Argument Span Extraction

Argument Span Extraction models that make use of the Connective and Argument 2 Labels are trained on reference annotation. Even though, the performance of the upstream models (*Discourse Connective Detection* and *Argument Position Classification*) is relatively high compared to the *Argument Span Extraction* models, there is still error propagation.

For the *Argument Span Extraction* of explicit relations the search space is limited to a single sentence; thus, all multi sentence arguments are missed. This constraint has a little effect on *Argument 2* spans. However, since as a candidate for inter-sentential *Argument 1* we use only immediately preceding sentence, together with this constraint we miss 12% of relations. Thus, detection of *Argument 1* spans of inter-sentential relations is a hard task, additionally due to the fact that there is no other span (connective or Argument 2) to delimit it. Even though we have trained CRF models for the task, previous sentence heuristic was performing with insignificant difference. Thus, the heuristic was selected for the submitted version, and it was augmented with the removal of sentence initial and final punctuation. For *Argument 2* of inter-sentential relations performance of CRF models is acceptably high (≈ 0.80).

The span of *Argument 2* of intra-sentential relations is the easiest to detect, since it is syntacti-

cally attached to the connective; and performances are high (≈ 0.89 on the development set using the features in Table 1). Thus, its output is used as a feature for *Argument 1* extraction. Interesting fact is that POS-tags have a negative effect on the *Argument 1 Span Extraction*.

4.5 Non-Explicit Relation Detection

Based on the output of *Argument Position Classification* a set of adjacent sentence pairs is generated as candidates for non-explicit discourse relations: Implicit, AltLex, and EntRel. For training the classification models we have generated No-Relation pairs using reference annotation, excluding all the sentences involved in inter-sentential relations (some relations have multiple sentence arguments). Additionally, since arguments of non-explicit relations are stripped of leading and trailing punctuation, the No-Relation pairs were preprocessed. The task of detecting relations proved to be hard.

Similar to *Connective Sense Classification* we attempted (1) flat classification into all PDTB senses + No-Relation (i.e. merging the task with *Relation Sense Classification* described in Section 4.6) and (2) hierarchical – first detect the presence of a relation then classify it into the hierarchy of senses. For the hierarchical detection of Non-Explicit relations we tried (1) Relation vs. No-Relation classification and (2) classification into relation types (Implicit, AltLex, EntRel) + No-Relation. The model that has the highest F-measure for actual relations turned out to be binary Relation vs. No-Relation classification (0.6988). However, since in the testing mode we don't have access to argument span information the performance is expected to drop significantly. The most robust feature combination for the task is Cartesian product of Brown Clusters of all the tokens from both arguments and Cartesian product of Brown Clusters of predicate, subject and direct and indirect objects (4 and 8 from Section 3.2).

4.6 Relation Sense Classification

After a sentence pair is classified as a relation, it is further classified into the hierarchy of senses. The models are trained on all the features from Section 3.2, excluding prefixed Bag-of-Words and Cartesian product of all tokens. Relations are classified directly into 14 PDTB senses + EntRel.

The task is extremely hard, the classification accuracy is 0.3899 and the model misses infrequent senses. Table 2 lists the captured senses with their percentages in training data and F-measures on the development set. The distribution of senses has a direct effect on its F-measure.

Sense	**%**	F_1
Expansion.Conjunction	19.0	0.4247
Expansion.Restatement	14.4	0.3212
Contingency.Cause.Reason	12.2	0.2945
Comparison.Contrast	9.5	0.0980
Contingency.Cause.Result	8.6	0.0563
Expansion.Instantiation	6.5	0.1918
Temporal.Asynchronous.Precedence	2.7	0.1290
Less Frequent and Partial Senses	4.1	0.0000
EntRel	23.1	0.5730
All (micro-average)	–	0.3899

Table 2: F-measures of non-explicit relation sense classification per sense, ordered by frequency in the training set.

The performances reported so far are on a specific task without error propagation from the upstream tasks. In the next section we report official Shared Task evaluation results.

5 Official Evaluation Metrics and Results

The official evaluation of CoNLL 2015 Shared Task on Shallow Discourse Parsing is done on a per-discourse relation basis. A relation is considered to be predicted correctly if the parser correctly identifies (1) discourse connective span (head), (2) spans and labels of both arguments, and (3) sense of a relation. The predicted connective and arguments spans have to match the reference spans *exactly*. Consequently, to get a true positive for a relation the parser has to get true positive on all the subtasks.

The task organizers also provided the evaluation script that reported precision, recall and F-measures for *Discourse Connective Detection*, joint *Sense Classification* scores for explicit and non-explicit relations, and joint *Argument Span Extraction* score for explicit and non-explicit relations. For argument spans three scores were reported: *Argument 1* and *Argument 2* individually and jointly. For *Sense Classification* the script reported performance on each of the senses and their macro-average. Later, performances for explicit and non-explicit relations were split. The participants had to evaluate their systems on 3 data sets: (1) Development (WSJ Section 22), (2) Test (WSJ Section 23), and the blind test set annotated specifically for the Shared Task.

The performance of our parser on each of the

	Explicit			Non-Explicit			All Relations		
Task	**Dev**	**Test**	**Blind**	**Dev**	**Test**	**Blind**	**Dev**	**Test**	**Blind**
Connective	0.9219	0.9271	0.8992	–	–	–	0.9219	0.9271	0.8992
Arg1	0.5646	0.5008	0.4903	0.4586	0.4437	0.4329	0.5225	0.4775	0.4654
Arg2	0.7748	0.7616	0.7068	0.4912	0.4744	0.5657	0.6230	0.6068	0.6260
Arg1&2	0.5075	0.4460	0.3959	0.4000	0.3730	0.3831	0.4499	0.4065	0.3886
Sense	0.4573	0.3260	0.2522	0.0601	0.0678	0.0681	0.3121	0.2526	0.1887
Parser	0.4760	0.3956	0.2997	0.1577	0.1330	0.1577	0.3055	0.2536	0.2184

Table 3: Task-level and parser-level F-measures of the parser on the development, test, and blind test sets for explicit and non-explicit relations individually and jointly. The Sense values are macro-averages.

Team	**P**	**R**	**F1**
lan15	0.2369	0.2432	0.2400
stepanov15	0.2094	0.2283	0.2184
li15b	0.1981	0.1737	0.1851

Table 4: Parser-level precision (P), recall (R), and F-measures (F1) of the submitted system on the blind test set. UniTN system is in **bold**.

metrics (tasks) per evaluation set is reported individually and jointly for explicit and non-explicit relations in Table 3. From the results, it is clear that non-explicit *Relation Sense Classification* is the hardest task. The next hardest task is intersentential *Argument 1 Span Extraction*. According to the organizers, the development, test and blind test sets are coming from the same domain. However, we observe a gradual decline in performance from development to test and from test to the blind test sets for each of the tasks on explicit relations. For non-explicit relations, on the other hand, performances vary and in some cases the performance on the blind test set is the highest (*Argument 2* spans).

The parser ranked the second on the test and the blind test sets and the third on the development set. For the comparison we also report performances of the systems ranked the first and the third in Table 4. The global F-measure of our parser on the blind test set is 0.2184, which is 0.0219 points lower than the first ranked system and 0.0333 points higher than the next best system. Comparing the performance with all the participants, we have observed that our parser maintains higher recall across the subtasks.

6 Conclusion

In this paper we have presented University of Trento parser submitted to CoNLL 2015 Shared Task on Shallow Discourse Parsing. We have described the discourse parsing architecture and models for each of the subtasks. The subtasks are categorized into span detection and classification. The span detection tasks are for explicit relations – Discourse Relation Detection and Argument Span Extraction; they are cast as token-level sequence labeling using Conditional Random Fields and argument span decisions are conditioned on relations' being intra- or inter-sentential. Classification tasks – Connective Sense Classification, Argument Position Classification, Non-Explicit Relation Detection, and Non-Explicit Relation Sense Classification – employ AdaBoost algorithm.

Participation in the CoNLL 2015 Shared Task on Shallow Discourse Parsing gave the teams a unique opportunity to compare their discourse parsing approaches on the same training and testing splits and the same automatic features. Even though the ranking of submitted systems depends on performances of all the modules, we can conclude that token-level sequence labeling for *Argument Span Extraction* of explicit discourse relations is a viable approach.

Participation additionally allowed us to identify potential points of improvement for our parser. For example, even though Discourse Connective Detection as sequence labeling has an F-measure of 0.8992 on the blind test set, it ranks 4th. Since it is the first step in the pipeline, increasing the robustness of the model is essential.

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.

Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. 2011. Shallow discourse parsing with conditional random fields. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*.

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.

Emily Pitler and Ani Nenkova. 2009. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP Conference*, pages 13–16.

Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 683–691.

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.

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)*.

Attapol T. Rutherford and Nianwen Xue. 2014. Discovering implicit discourse relations through Brown Cluster pair representation and coreference patterns. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*.

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

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

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

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*.