Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 1103–1109
Marseille, 11–16 May 2020

Semi-Supervised Tri-Training for Explicit Discourse Argument Expansion

René Knaebel, Manfred Stede
Applied Computational Linguistics
Department of Linguistics
University of Potsdam
Germany
{rknaebel,stede}@uni-potsdam.de

Abstract

This paper describes a novel application of semi-supervision for shallow discourse parsing. We use a neural approach for sequence tagging and focus on the extraction of explicit discourse arguments. First, additional unlabeled data is prepared for semi-supervised learning. From this data, weak annotations are generated in a first setting and later used in another setting to study performance differences. In our studies, we show an increase in the performance of our models that ranges between 2-10 % F1 score. Further, we give some insights to the generated discourse annotations and compare the developed additional relations with the training relations. We release this new dataset of explicit discourse arguments to enable the training of large statistical models.

Keywords: discourse parsing, pdtb, bilstm, semi-supervision, tri-training

1. Introduction

Discourse relations, holding between neighbouring spans of text, are known to play a central role for explaining why a text is coherent, and how it is to be interpreted. A subset of these relations is signaled by specific words, so-called discourse connectives (or discourse markers or cues), and thus referred to as explicit discourse relations. Discourse relations in general and their understanding are important for tasks such as machine translation, abstractive summarization, and text simplification. Short examples of such explicit discourse relations (here within the same sentence) are:

- He ran to the school **because** it was raining.
- **Although** she took her bike, she came late.
- **If** things work out, **then** everybody will be happy.

Shallow discourse parsing (SDP) (Lin et al., 2014) is the research area that builds models to uncover those discourse relations within texts. SDP consists of the main tasks of identifying connectives, demarcating their arguments, assigning senses to them, and finding the senses of so-called *implicit relations* holding between adjacent text spans without a lexical signal being present. In this work, however, we focus on explicit discourse relations only, and further, we leave the sense selection aside, as relation span labeling can be handled independently of a relation's sense.

Because of its complexity, it is hard to get annotated data for training statistical models to perform SDP. Therefore, our goal is to produce high-quality annotations other than the standard corpus used in the field—the *Penn Discourse Treebank* (PDTB) (Prasad et al., 2008)—in order to improve performance on explicit argument extraction.

Learning from unlabeled data as in unsupervised learning still remains a challenging problem. The branch which combines supervised and unsupervised methods is referred to as semi-supervised learning. In particular, in this paper, we focus on learning from unlabeled data by producing proxy labels.

This is a hard problem in shallow discourse argument extraction because the proposed models do not work very well. Therefore, we adopt two variations of the same problem. The first is slightly easier and better suited for labeling new data. Thereafter, we study the noisy training data thus produced with a more complex model variation. Also, we will analyze the extracted data and compare certain statistics to the original training data.

Specifically, we tackle the problem of limited training data by adapting existing models of our previous work and increasing the amount of training data by using the output of their predictions. As these predictions are not perfect, the resulting data is noisy and biased by the model's architecture. Our aim is to study the influence on a specific model when the training includes additional noisy argument labels. So, our goal is not to improve the architectural design of models for SDP *per se*, but to increase the amount of available data for training these models. Given this purpose, in this paper we will not compare various state-of-the-art SDP components but just select one particular architecture and demonstrate the effects of data augmentation on that architecture.

The contributions of this paper are summarized as follows:

1. We design extensive experiments on the task of extracting explicit discourse relations using additional unlabeled data. We use two separate phases for training that handle tasks of different complexity. The results show a promising increase in performance.

2. The amount of explicit discourse relations extracted from the additional data makes the available training data twice as large. The resulting dataset will be publicly available.

In the following, Section 2. discusses relevant related work, Section 3. describes the unlabeled corpus, and Section 4. explains our method. The experiments and results are presented in Section 5., followed by conclusions in Section 6..

2. Related Work

In this section, we first discuss some relevant work in the area of discourse parsing and on the model used in this paper. Next, an overview of semi-supervised learning is provided as well as other work that deals with the problem of data sparsity.

2.1. Shallow Discourse Parsing

Shallow discourse parsing is a challenging task, which was promoted by the development of the second version of the Penn Discourse Treebank (PDTB2) (Prasad et al., 2008) and further adapted by the shared tasks at CoNLL 2015 and 2016 (Xue et al., 2015; Xue et al., 2016). Several systems have been proposed at the competitions (e.g., (Wang et al., 2015; Wang and Lan, 2016; Oepen et al., 2016)), and they largely follow the pipeline model of Lin et al. (2014), which consists of successive tasks of connective identification, argument labeling, and sense classification for both explicit and implicit relations.

Argument labeling with recurrent neural networks was done first by Wang et al. (2015) in their DCU parser. In addition to word embeddings, they also used other features, such as POS tags, syntactic relations, and lexical features. They distinguish between intra-sentential and inter-sentential relations (Ghosh et al., 2011), and thus train separate models for each individual labeling task. In contrast, the approach used for this paper does not rely on sentence boundaries and uses word embeddings only.

Recently, fixed sized windows were introduced with neural networks for argument labeling as a 4-class sequence classification task. First, Hooda and Kosseim (2017) studied explicit argument extraction on predefined windows with a fixed length corresponding to the maximum span length of arguments. Then, in our previous work, we adapted their approach and described a more general procedure that integrates connective classification into the process of argument extraction by using moving windows over a discourse (Knaebel et al., 2019). This approach was limited by the relatively small amount of data available in the PDTB2.

Malmi et al. (2018) also improve data availability but focus on connective prediction and further limit themselves to special cases with two consecutive sentences where the connective is at the beginning of the second sentence. Our approach instead applies to the full range of spans of explicit discourse relations, but it is limited by the model's window size used to predict a relation.

2.2. Semi-Supervised Learning

Semi-supervised learning is a research area that tries to jointly learn from labeled and unlabeled data. Many different ways have been proposed to tackle this challenge.

Clark et al. (2018) combine their main task with additional auxiliary tasks such as predicting masked words from context to gain performance. Also, semi-supervised learning is used on unlabeled data to identify co-occurring features (Hernault et al., 2010a; Hernault et al., 2010b). These authors define auxiliary training tasks to improve model performance, but the unlabeled data is restricted to individual text spans that contain single relations.

Corpus	docs	length	tokens	relations
train	1,756	22.28	933,049	14,722
dev	79	21.35	39,712	680
test	91	25.67	55,453	923
blind	71	18.95	34,621	556
bbc-news	2,223	18.60	958,212	*19,576
bbc-sport	736	17.29	284,528	*8,186

Table 1: General statistics to contrast labeled and unlabeled data. Both sets have comparable sizes. (*) The number of relations for unlabeled data is approximated by a connective classifier.

There are various approaches for bootstrapping and self-training with neural models. In *Self-Training* (McClosky et al., 2006; Yarowsky, 1995), predictions of the same model are used with respect to the model's confidence in a particular prediction. Despite its simple mechanism, this algorithm comes with a high bias, which is unfavorable for learning new directions within the data. In contrast, *Multi-View-Training* (Zhou and Goldman, 2004; Søgaard, 2010) tries to compensate this bias by different views of the data. These different views may be approached by separate feature sets, data splits, and models. In a recent study, Ruder and Plank (2018) show that *Tri-Training* (Zhi-Hua Zhou and Ming Li, 2005), a form of Multi-View-Training with three independently trained models, should be considered as a strong baseline for neural semi-supervised learning.

Recently, Chen et al. (2018) use a network architecture called Tri-net that works similar to ours. Their experiments deal with the task of image classification, though.

3. Data Collection

For additional data, we consider the work of Greene and Cunningham (2006) who present the *BBC Datasets*. This data collection consists of two parts, *bbc-news* and *bbc-sport*, both from the years 2004 to 2005. The first dataset contains 2225 documents including 5 categories (business, entertainment, politics, sport, tech). The second dataset focuses on sports articles and contains 737 documents distinguishing 5 categories, too (athletics, cricket, football, rugby, tennis). As shown in more detail in Table 1, the number of documents for BBC outnumbers the original PDTB training data. Correspondingly, the approximate number of explicit relations (estimated by the number of predicted connectives) in the BBC corpus is quite high in comparison with the PDTB. This must be taken into account for training the model on these proxy labels of the additional data. Because the predicted annotations are not as reliable as the gold annotations, the model could easily make false assumptions about the data when using a bad balance of gold labels and proxy labels. Throughout this work, we report both parts of the BBC data separately.

3.1. Preprocessing

We preprocess the raw BBC dataset[1] such that the format is comparable to the provided PDTB (CoNLL-format)

[1] http://mlg.ucd.ie/datasets/bbc.html

dataset. First of all, for each document in a corpus, we extract the main document text. After normalization, we use the Penn Treebank Tokenizer, implemented in NLTK[2], for similarity. We further process each tokenized document using Spacy[3] and use their part-of-speech tags and dependency trees. For generating constituency trees, we use an additional module provided by the Benepar project(Kitaev and Klein, 2018). Except for the normalized tokens, predicted information is only used for traditional models and left out for the neural models. We publish all our scripts and findings for future research.[4]

4. Method

The main goal of our work is to consider unlabeled data for training and, additionally, to produce new automatically annotated data that can be used in other settings as well. Our focus is on explicit discourse relations and in particular identifying argument spans of those relations, also called explicit argument labeling.

We split this general task into two phases, where the problem of generating new data is made easier for better predictions. In the first phase, we use a combination of a traditional feature-based connective classifier and a neural model for argument extraction. By using a separate connective model first, the window for the neural model is already determined. Thus, we reduce lots of the complexity of the more general explicit argument extraction task (empty windows or those where relations are not centered). We train this whole model on the annotated data and use its predictions on the unlabeled data for the second training phase afterward. Our approach follows the idea of multi-view models—training with different aspects to reduce model bias.

In the second phase, we evaluate our additional data in a more complex training setting. For this, we use a neural model that can jointly predict connectives and their arguments. The model predicts the occurrence of a relation, and for each relation, labels every token in a window as to its role (Arg1, Arg2, Connective). Finally, we compare models trained with and without the additional relations, to examine their effect on the training.

The overall structure of our experiment is shown in Figure 1.

4.1. Neural Argument Extraction

For argument extraction, we adapt our previous work on window-based neural models for discourse relation extraction (Knaebel et al., 2019). There, we describe tasks of different complexity that the neural model is being trained on. For our experiments, we use two of these settings; one model is trained to extract argument spans around previously extracted connectives, and another one is trained to jointly extract connectives and their arguments.

The *Neural Connective Argument Extractor* (`NCA`) is a two-component pipeline that consists of a traditional connective classifier (Pitler and Nenkova, 2009) and a neural model. The prediction of the first component is used to identify the connective. A fixed-sized window is placed over this area such that the connective is centered within the model's window. Then, each token is classified as being part of one of the argument.

The *Neural Explicit Argument Extractor* (`NEA`) solves a more complex task where the connective is not given by an external model. Instead, the model is trained to determine the presence or absence of relations. A sliding window approach is used, where each token within a window is assigned to one of four classes (None, Arg1, Arg2, Connective). The individual predictions for each document are aggregated and define the final set of relations.

The fundamental component in both extractors is the neural model for sequential classification. We adjust the original topology of the neural network to increase the model's performance. Nevertheless, the approach of our previous work is independent of the model's topology. As shown in Figure 2, we add spatial dropout (Tompson et al., 2014) after the embedding layer, and we double the recurrent layer.

As described in Section 5., we also adjust sizes of hidden and recurrent layers for both types of models.

4.2. Full Explicit Relation Extraction

Semi-supervised learning is intermediate between supervised learning and unsupervised learning. The idea is to use additionally unlabeled data to support the model's training on labeled data. In *multi-view training*, several models are trained on the same task but with different views on the data. Thus, they should complement each other's predictions by improving performance and reducing the overall bias.

In our experiments (see Phase 1 in Figure 1), we use *tri-training* (Zhi-Hua Zhou and Ming Li, 2005), a multi-view technique where three independent models are used to balance each other's predictions. Following the authors, the most common way to achieve diversity is by bootstrapping the training data. After each training round, a new bootstrap is generated per model, taking recently extracted relations into account. Bootstraps are sampled over possible documents, with the effect that model views are fully independent of each other, without overlapping relations. Further, we achieve different inductive bias of all three models by varying the models' architectures as proposed by Zhou and Goldman (2004). A prediction is considered reliable and taken into account for future training if at least two models fully agree on a sample's prediction.

5. Experiments

As described above and sketched out in Figure 1, we design two consecutive phases for training. The first phase is a simpler problem and we use it for the model's adaption to the new data and the extraction of new explicit relations. During the second phase, we study the quality of the extracted relations from the former phase, by training a more complex model with and without the developed training data.

Since we cannot fully reconstruct the evaluation of the CoNLL Shared Task (as certain tokens were not taken into account for scoring), we adapt the scheme used there and show different gradations of the exact match scoring. In particular, we count argument spans as correctly classified

[2] `www.nltk.org`

[3] `www.spacy.io`

[4] `https://github.com/rknaebel/bbc-discourse`

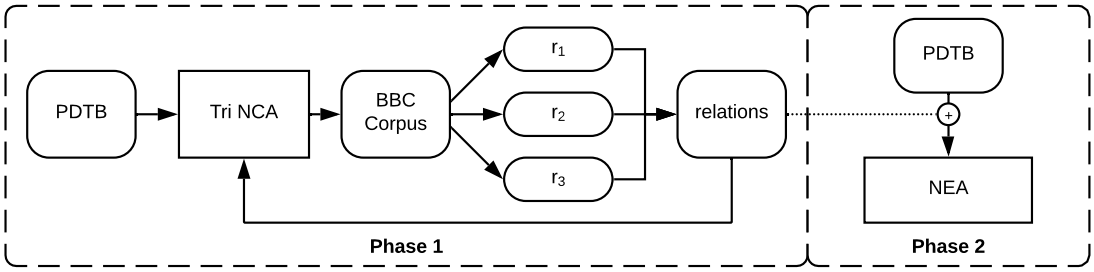

Figure 1: Experiments separated in two phases. The first is used to produce high quality proxy labels. Therefore, each of the three models predicts a relation (r_1, r_2, r_3) on a window. The relation is considered valid (and used as proxy label), if any two models agree; the second phase integrates proxy labels into training.

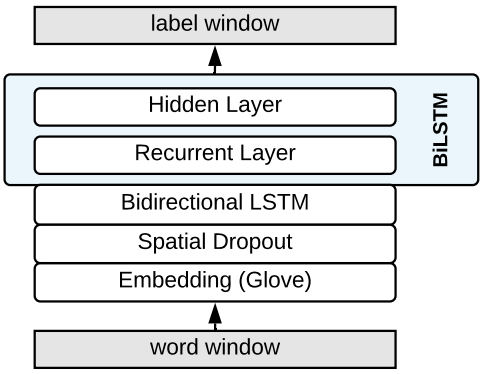

Figure 2: Neural architecture used throughout the paper. Adapts the former model (Knaebel et al., 2019) and highlights changes (additional dropout and bilstm layer).

if they have a certain overlap with the reference span. For the overlap score, we first compute precision and recall between span prediction and its reference. Then, we use the resulting F1 score as the threshold for regarding the overlap of two spans as correct. Specifically, for our evaluations we use several thresholds ($t \in \{0.7, 0.8, 0.9\}$).
Each experiment (window length) is run four times to give a more reliable overview of the results. Thus, the F1 scores shown in our result tables are averages of the corresponding runs. As is commonly done, we evaluate each part of a discourse relation separately (Conn, Arg1, Arg2) and also the concatenation of both arguments (referred to as Both). The upper three rows of each of these tables show the baseline's performance. In comparison, the final iteration's results are demonstrated in the lower three rows.

5.1. Phase 1: Relation Extraction

In the first part of our experiments, we use the Tri-model (see Section 4.2.) and train it on the annotated data. We use the different parts of PDTB2 for `training`, `validation`, and `test`, respectively. Throughout all experiments, `test` and `validation` are untouched. The former is used for overall evaluation, while the latter prevents overfitting.
The tri-model's connective classifier is first trained only on `training`. Then, for each neural argument extractor, we bootstrap 70% of the available training data for having a different data bias per model. Also, each model differs in its architecture as we choose varying values for the hidden layer and the recurrent layer. More precisely, for each neural model m_i in the tri-model, we choose respective values $(h_i, r_i) \in \{(256, 512), (512, 256), (512, 512)\}$. For the experiments, we study varying window lengths, $w \in \{50, 100, 150\}$.
The results in Table 2 already indicate an increase of performance from baseline to final for most of the entries.

5.2. Phase 2: Data Adaption

In the second part of our experiments, we use the explicit discourse relations generated during training the tri-model on unlabeled data. For this experiment, we use the more complex training problem where a single neural model is trained to jointly predict a connective and its arguments. Also, the model must distinguish between the presence and absence of any explicit relation.
As baseline, first, the `NAE` is trained only on PDTB annotations (`training`). Then, we use the same model configuration, but train it on all available data (PDTB+BBC). The results in Table 3 show the evaluation on `test`.
Additionally, we evaluate the `NAE` on CoNLL's `blind` dataset (see Table 4), which comes from a different source (Wikipedia). The performance of our model is lower than for `test`, but still records an increase of performance by transferring additional weakly annotated data through training. This also means that data from other sources would probably be beneficial for this more challenging evaluation.

5.3. BBC Relation Analysis

In this section, we briefly summarize the average properties of extracted relations recorded for the final runs of our first phase experiments. We study two items of meta-information that help to compare the extractions with the labeled data relations.
In Table 5, we compare the argument lengths by computing the average spans of `Arg1`, `Arg2`, and `Both`. We see that relations from `training` are quite long, in comparison with any extracted relation set, which is caused by a few outliers in the PDTB. A reason that the extracted relations do not contain any such long relation is the limited

Threshold	0.7				0.8				0.9			
Baseline	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both
w50	92.72	63.93	84.11	72.20	92.72	55.81	78.30	62.83	92.72	47.33	70.97	48.77
w100	90.86	57.76	80.03	68.65	90.86	48.55	75.15	56.70	90.86	41.56	67.99	43.48
w150	93.13	54.80	81.13	65.43	93.13	44.94	75.95	53.63	93.13	36.78	67.84	39.08
Final												
w50	92.74	65.60	84.42	73.25	92.74	58.07	79.18	63.62	92.74	50.06	72.51	50.85
w100	92.67	62.28	83.87	71.97	92.67	53.33	79.82	61.27	92.67	55.49	73.85	48.97
w150	93.16	56.72	82.79	66.95	93.16	46.39	78.42	54.86	93.16	38.17	71.11	40.65

Table 2: Results of the NCA Tri-model for each window size averaged over corresponding runs on `test`. Comparison between baseline and final iteration. Exact match with varying overlap threshold.

Threshold	0.7				0.8				0.9			
Baseline	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both
w50	62.45	43.43	62.43	51.14	62.29	35.42	59.20	41.21	62.20	28.63	53.79	30.94
w100	62.39	44.46	61.78	51.95	62.21	36.44	58.39	42.52	62.14	29.59	53.00	31.39
w150	59.96	42.57	60.85	49.90	59.68	34.36	57.33	39.92	59.68	27.71	51.17	29.52
Final												
w50	64.15	47.93	63.84	54.33	63.95	40.65	60.60	45.95	63.80	35.13	55.49	36.83
w100	68.85	52.17	68.56	60.23	68.69	44.07	65.23	51.35	68.60	38.17	60.25	40.48
w150	67.00	47.97	66.47	55.89	66.84	39.21	63.09	46.38	66.77	32.11	57.66	35.12

Table 3: Results of the NEA for each window size averaged over corresponding runs on `test`. Comparison between baseline and final iteration. Exact match with varying overlap threshold.

span length (window size), which is caused by the window-based approach. Also, a model's performance concerning the distance between predicted spans and true spans does not monotonically improve by increasing window length but also overfits at some point (as shown by the bbc-news-150 for individual argument prediction).

In Table 6, we study differences in the position of `Arg1` compared to `Arg2`. Therefore, we define four categories (referring to the relative position of `Arg1`) and sort each relation into one of them. Simple situations contain arguments where one argument fully precedes the other one (`prev` and `next`). The remaining situations are those where one argument is embedded in the other one. If `Arg1` is surrounded by `Arg2`, we count it as `inside` and, otherwise, it counts as `outside`. Other (more complex) constellations do not exist per definition.

All three models have a strong prediction bias to the dominating argument position in `training`. The overall relation between the category sizes remains in the models' predictions (`prev` < `next` < `inside` < `outside`).

6. Conclusions and Future Work

In this work, we developed a two-phase semi-supervised learning setting to improve explicit discourse argument extraction using unlabeled data. We first designed a simpler problem to produce high-quality annotations for this additional data. Then, the extracted relations were used in the more complicated setting, without knowing exact positions of explicit relations. With our work, we showed the positive effect of additional data for neural discourse relation extraction, even though the data was not perfectly labeled. Finally, we studied the characteristics of the weak extractions produced by our method and compared them to distributions of the human-annotated corpus. We published these explicit discourse arguments to help future research in this area.

We adapted a neural argument extraction architecture and made small changes to improve the initial performance. These changes were necessary to make the baseline's predictions helpful during the training process.

There is currently not much work on semi-supervised learning for sequence prediction. Therefore it was difficult to decide for good criteria to merge predictions into training data. Also, our reliability condition (two models fully agree) is probably too strong for sequences of up to 150 elements.

In future work, we plan to continue addressing the same genre of data (news articles) but want to study model behavior also with domain shifting, using texts from literature or speeches, for example. At least a few human-annotated samples must be made available for evaluating these unseen domains.

Further, our present work is limited to explicit discourse arguments, but it would be interesting to generalize this approach to cover implicit arguments, too. The position of implicit relations in PDTB2 is limited to consecutive sentences, but the task still remains challenging for a single neural model because of the small amount of training data available.

7. Acknowledgments

This research was supported by the Federal Ministry of Education and Research (BMBF), Contract 01IS17059.

8. Bibliographical References

Chen, D.-D., Wang, W., Gao, W., and Zhou, Z.-H. (2018). Tri-net for semi-supervised deep learning. In *Proceed-*

Threshold	0.7				0.8				0.9			
Baseline	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both	Conn	Arg1	Arg2	Both
w50	59.42	43.59	58.44	51.78	59.24	35.52	55.47	41.92	59.20	29.48	51.75	30.45
w100	59.26	44.26	58.12	52.31	59.01	36.27	55.29	42.59	58.97	29.71	52.06	30.28
w150	56.01	42.18	57.01	50.26	55.70	34.13	53.97	40.27	55.70	27.89	50.07	28.74
Final												
w50	58.36	46.40	56.83	52.68	58.15	39.34	54.04	44.87	58.05	34.15	51.47	34.25
w100	61.69	47.26	60.29	54.44	61.38	39.65	57.60	44.72	61.24	33.91	54.97	34.06
w150	59.91	42.96	58.18	50.99	59.69	34.81	55.33	40.30	59.59	28.52	52.10	29.54

Table 4: Results of the NEA for each window size averaged over corresponding runs on `blind`. Comparison between baseline and final iteration. Exact match with varying overlap threshold.

Corpus	Both	Arg1	Arg2
train	40.10	16.53	13.28
bbc-news-50	27.72	12.58	12.62
bbc-sport-50	26.22	11.74	12.00
bbc-news-100	32.61	16.27	13.90
bbc-sport-100	29.38	14.02	12.89
bbc-news-150	36.35	20.03	14.02
bbc-sport-150	32.38	16.97	13.05

Table 5: Relation Spans. Contrasts the length of arguments between labeled and unlabeled training data.

Corpus	prev	next	inside	outside
train	12630.00	1433.00	632.00	27.00
bbc-news-50	17051.25	876.50	262.50	2.50
bbc-sport-50	7454.25	242.75	73.00	1.00
bbc-news-100	16919.50	807.75	216.00	3.00
bbc-sport-100	7428.00	224.25	54.00	1.75
bbc-news-150	15672.25	759.00	317.00	7.00
bbc-sport-150	6887.75	210.50	87.25	3.25

Table 6: First Argument Position: Contrasts labeled and unlabeled data distribution with respect to the position of the first argument compared to the second one.

ings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pages 2014–2020. International Joint Conferences on Artificial Intelligence Organization, 7.

Clark, K., Luong, M., Manning, C. D., and Le, Q. V. (2018). Semi-supervised sequence modeling with cross-view training. *CoRR*, abs/1809.08370.

Ghosh, S., Johansson, R., Riccardi, G., and Tonelli, S. (2011). Shallow discourse parsing with conditional random fields. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1071–1079. Asian Federation of Natural Language Processing.

Greene, D. and Cunningham, P. (2006). Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proc. 23rd International Conference on Machine learning (ICML'06)*, pages 377–384. ACM Press.

Hernault, H., Bollegala, D., and Ishizuka, M. (2010a). A semi-supervised approach to improve classification of infrequent discourse relations using feature vector extension. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 399–409, Cambridge, MA, October. Association for Computational Linguistics.

Hernault, H., Bollegala, D., and Ishizuka, M. (2010b). Towards semi-supervised classification of discourse relations using feature correlations. In *Proceedings of the SIGDIAL 2010 Conference*, pages 55–58, Tokyo, Japan, September. Association for Computational Linguistics.

Hooda, S. and Kosseim, L. (2017). Argument labeling of explicit discourse relations using lstm neural networks. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 309–315. INCOMA Ltd.

Kitaev, N. and Klein, D. (2018). Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia, July. Association for Computational Linguistics.

Knaebel, R., Stede, M., and Stober, S. (2019). Window-based neural tagging for shallow discourse argument labeling. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 768–777, Hong Kong, China, November. Association for Computational Linguistics.

Lin, Z., Ng, H. T., and Kan, M.-Y. (2014). A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, 20:151–184.

Malmi, E., Pighin, D., Krause, S., and Kozhevnikov, M. (2018). Automatic prediction of discourse connectives. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

McClosky, D., Charniak, E., and Johnson, M. (2006). Effective self-training for parsing. In *HLT-NAACL*.

Oepen, S., Read, J., Scheffler, T., Sidarenka, U., Stede, M., Velldal, E., and Øvrelid, L. (2016). Opt: Oslo–potsdam–teesside. pipelining rules, rankers, and classifier ensembles for shallow discourse parsing. In *Proceedings of the CoNLL-16 shared task*, pages 20–26. Association for Computational Linguistics.

Pitler, E. and Nenkova, A. (2009). Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 13–16, Suntec, Singapore, August. Associa-

tion for Computational Linguistics.

Prasad, R., Dinesh, N., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A., and Webber, B. (2008). The penn discourse treebank 2.0. In *In Proceedings of LREC*.

Ruder, S. and Plank, B. (2018). Strong baselines for neural semi-supervised learning under domain shift. *CoRR*, abs/1804.09530.

Søgaard, A. (2010). Simple semi-supervised training of part-of-speech taggers. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 205–208, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., and Bregler, C. (2014). Efficient object localization using convolutional networks. *CoRR*, abs/1411.4280.

Wang, J. and Lan, M. (2016). Two end-to-end shallow discourse parsers for english and chinese in conll-2016 shared task. In *Proceedings of the CoNLL-16 shared task*, pages 33–40. Association for Computational Linguistics.

Wang, L., Hokamp, C., Okita, T., Zhang, X., and Liu, Q. (2015). The dcu discourse parser for connective, argument identification and explicit sense classification. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 89–94. Association for Computational Linguistics.

Xue, N., Ng, H. T., Pradhan, S., Prasad, R., Bryant, C., and Rutherford, A. (2015). The conll-2015 shared task on shallow discourse parsing. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 1–16. Association for Computational Linguistics.

Xue, N., Ng, H. T., Pradhan, S., Rutherford, A., Webber, B., Wang, C., and Wang, H. (2016). Conll 2016 shared task on multilingual shallow discourse parsing. In *Proceedings of the CoNLL-16 shared task*, pages 1–19. Association for Computational Linguistics.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA, June. Association for Computational Linguistics.

Zhi-Hua Zhou and Ming Li. (2005). Tri-training: exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, 17(11):1529–1541, Nov.

Zhou, Y. and Goldman, S. (2004). Democratic co-learning. In *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, ICTAI '04, pages 594–202, Washington, DC, USA. IEEE Computer Society.