Implicit Discourse Relation Classification: We Need to Talk about Evaluation

Najoung Kim* Department of Cognitive Science Johns Hopkins University n.kim@jhu.edu

Abstract

Implicit relation classification on Penn Discourse TreeBank (PDTB) 2.0 is a common benchmark task for evaluating the understanding of discourse relations. However, the lack of consistency in preprocessing and evaluation poses challenges to fair comparison of results in the literature. In this work, we highlight these inconsistencies and propose an improved evaluation protocol. Paired with this protocol, we report strong baseline results from pretrained sentence encoders, which set the new state-of-the-art for PDTB 2.0. Furthermore, this work is the first to explore fine-grained relation classification on PDTB 3.0. We expect our work to serve as a point of comparison for future work, and also as an initiative to discuss models of larger context and possible data augmentations for downstream transferability.

1 Introduction

Understanding discourse relations in natural language text is crucial to end tasks involving larger context, such as question-answering (Jansen et al., 2014) and conversational systems grounded on documents (Saeidi et al., 2018; Feng et al., 2020). One way to characterize discourse is through relations between two spans or *arguments* (ARG1/ARG2) as in the Penn Discourse TreeBank (PDTB) (Prasad et al., 2008, 2019). For instance:

[*Arg1 I live in this world*,] [*Arg2 assuming that there is no morality, God or police.*] (wsj_0790) Label: EXPANSION.MANNER.ARG2-AS-MANNER

The literature has focused on implicit discourse relations from PDTB 2.0 (Pitler et al., 2009; Lin et al., 2009), on which deep learning has yielded substantial performance gains (Chen et al., 2016; Liu and Li, 2016; Lan et al., 2017; Qin et al., 2017; Bai and

*Work done while at IBM Research.

Song Feng, Chulaka Gunasekara, Luis A. Lastras IBM Research AI {sfeng@us, chulaka.gunasekara@, lastrasl@us}.ibm.com

Zhao, 2018; Nguyen et al., 2019, *i.a.*). However, inconsistencies in preprocessing and evaluation such as different label sets (Rutherford et al., 2017) pose challenges to fair comparison of results and to analyzing the impact of new models. In this paper, we revisit prior work to explicate the inconsistencies and propose an improved evaluation protocol to promote experimental rigor in future work. Paired with this guideline, we present a set of strong baselines from pretrained sentence encoders on both PDTB 2.0 and 3.0 that set the state-of-the-art. We furthermore reflect on the results and discuss future directions. We summarize our contributions as follows:

- We highlight preprocessing and evaluation inconsistencies in works using PDTB 2.0 for implicit discourse relation classification. We expect our work to serve as a comprehensive guide to common practices in the literature.
- We lay out an improved evaluation protocol using section-based cross-validation that preserves document-level structure.
- We report state-of-the-art results on both toplevel and second-level implicit discourse relation classification on PDTB 2.0, and the first set of results on PDTB 3.0. We expect these results to serve as simple but strong baselines that motivate future work.
- We discuss promising next steps in light of the strength of pretrained encoders, the shift to PDTB 3.0, and better context modeling.

2 The Penn Discourse TreeBank (PDTB)

In PDTB, two text spans in a discourse relation are labeled with either one or two senses from a three-level sense hierarchy. PDTB 2.0 contains around 43K annotations with 18.4K explicit and 16K implicit relations in over 2K Wall Street Journal (WSJ) articles. Identifying implicit relations (i.e., without explicit discourse markers such as

Model	Ji	Lin	P&K	X-Accuracy
Majority class	26.18	26.11	28.54	26.42
Adversarial Net (Qin et al., 2017)	46.23	44.65	-	-
Seq2Seq+MemNet (Shi and Demberg, 2019)	47.83	45.82	-	41.29 [†]
ELMo (Bai and Zhao, 2018)	48.22	45.73	-	-
ELMo, Memory augmented (Bai et al., 2019)	49.15	46.08	-	-
Multitask learning (Nguyen et al., 2019)	49.95	46.48	-	-
BERT+MNLI (Nie et al., 2019)	-	-	53.7	-
BERT+DisSent Books 5 (Nie et al., 2019)	-	-	54.7	-
BERT (base, uncased)	52.13 (±0.50)	51.41 (±1.02)	52.00 (±1.02)	49.68 (±0.35)
BERT (large, uncased)	57.34** (±0.79)	55.07** (±1.01)	55.61 (±1.32)	53.37 (±0.22)
XLNet (base, cased)	54.73 (±1.26)	55.82*** (±0.79)	54.71 (±0.45)	52.98 (±0.29)
XLNet (large, cased)	61.29 *** (±1.49)	58.77 *** (±0.99)	59.90 * (±0.96)	57.74 (±0.90)

Table 1: Accuracy on PDTB 2.0 L2 classification. We report average performance and standard deviation across 5 random restarts. Significant improvements according to the $N - 1 \chi^2$ test after Bonferroni correction are marked with *,** ,*** (2-tailed p < .05, < .01, < .001). We compare the best published model and the median result from the 5 restarts of our models. Because we use section-based cross-validation, significance over [†] is not computed.

but) is more challenging than explicitly signaled relations (Pitler et al., 2008). The new version of the dataset, PDTB 3.0 (Prasad et al., 2019), introduces a new annotation scheme with a revised sense hierarchy as well as 13K additional datapoints.² The third-level in the sense hierarchy is modified to only contain asymmetric (or directional) senses.

2.1 Variation in preprocessing and evaluation

We survey the literature to identify several sources of variation in preprocessing and evaluation that could lead to inconsistencies in the results reported.

Choice of label sets. Due to the hierarchical annotation scheme and skewed label distribution, a range of different label sets has been employed for formulating classification tasks (Rutherford et al., 2017). The most popular choices for PDTB 2.0 are: (1) top-level senses (L1) comprised of four labels, and (2) finer-grained Level-2 senses (L2). For L2, the standard protocol is to use 11 labels after eliminating five infrequent labels as proposed in Lin et al. (2009). Sometimes ENTREL is also included in the L2 label set (Xue et al., 2015). Level-3 senses (L3) are not often used due to label sparsity.

Data partitioning. The variability of data splits used in the literature is substantial. This is problematic considering the small number of examples in a typical setup with 1-2 WSJ sections as test sets. For instance, choosing sections 23-24 rather than 21-22 results in an offset of 149, and a label offset as large as 71 (COMPARISON.CONTRAST).

This is a large enough difference to cast doubt on claims for state-of-the-art, considering the small size of the test sets (~ 1000). We illustrate the variability of split choices in published work in Appendix B. Recently, splits recommended by Prasad et al. (2008) and Ji and Eisenstein (2015) (*Ji*) are the most common, but splits from Patterson and Kehler (2013) (*P&K*), Li and Nenkova (2014), *i.a.*, have also been used. The Prasad et al. split is frequently attributed to Lin et al. (2009) (*Lin*), and thus we adopt this naming convention.

Multiply-annotated labels. Span pairs in PDTB are optionally annotated with multiple sense labels. The common practice is either taking only the first label or the approach in Qin et al. (2017), *i.a.*, where instances with multiple annotations are treated as separate examples during training. A prediction is considered correct if it matches any of the labels during testing. However, a subtle inconsistency exists even across works that follow the latter approach. In PDTB, two connectives (or inferred connectives for implicit relations) are possible for a span pair, where the second *connective* is optional. A connective can each have two semantic classes (i.e., the labels), where the second class is optional. Thus, a maximum of four distinct labels are possible for each span pair. However, in the actual dataset, the maximum number of distinct labels turns out to be two. An inconsistency arises depending on which of the four possible label fields are counted. For instance, Qin et al. (2017) treat all four fields (SCLASS1A, SCLASS1B, SCLASS2A, SCLASS2B; see link) as possible labels, whereas Bai and Zhao (2018); Bai et al. (2019) use only

 $^{^{2}}$ Note that there has been an update to PDTB 3.0 since this article has been written. This affects around 130 datapoints.

SCLASS1A,SCLASS2A. Often, this choice is implicit and can only be deduced from the codebase.

Random initialization. Different random initializations of a network often lead to substantial variability (Dai and Huang, 2018). It is important to consider this variability especially when the reported margin of improvement can be as small as half a percentage point (see cited papers in Table 1). We report the mean over 5 random restarts for existing splits, and the mean of mean cross-validation accuracy over 5 random restarts.³

3 Proposed Evaluation Protocol

While Xue et al. (2015) lay out one possible protocol, it does not fully address the issues we have raised in Section 2. Another limitation is the unavailability of the preprocessing code as of the date of this submission. We describe our proposal below, which will be accompanied by a publicly available preprocessing code.⁴ In addition to accounting for the variation previously discussed, we take Shi and Demberg (2017)'s concerns into consideration.

Cross-validation. We advocate using crossvalidation for L2 classification, sharing the concerns of Shi and Demberg (2017) on label sparsity. However, we propose using cross-validation at section-level rather than individual example-level as suggested by Shi and Demberg (2017). This is to preserve paragraph and document structures, which are essential for investigating the effect of modeling larger context (e.g., Dai and Huang 2018). We further illustrate the potential utility of document structure in Section 4. We suggest dividing the 25 sections of PDTB into 12 folds with 2 development, 2 test and 21 training sections in each fold. We used a sliding window of two sections starting from P&K (dev: 0-1, test: 23-24, train: 2-22). All but one section (22) is used exactly once for testing.

Whether future works should evaluate on these particular cross-validation splits or on randomized splits (Gorman and Bedrick, 2019) is an open issue; we provide an additional discussion in Appendix F.

Label sets. We recommend reporting results on both L1 and L2, using the standard 11-way classification for L2 in PDTB 2.0. A standardized label set

does not exist yet for L2 in PDTB 3.0 (L1 remains unchanged). We propose using only the labels with > 100 instances, which leaves us with 14 senses from L2 (see Appendix A for counts). We suggest using all four possible label fields if the senses are multiply-annotated, as discussed in Section 2.1.

Model	X-Accuracy $(\pm \sigma)$
Majority class	26.61
BERT (base, uncased) BERT (large, uncased) XLNet (base, cased) XLNet (large, cased)	$\begin{array}{c} 57.60 \ (\pm 0.19) \\ 61.02 \ (\pm 0.19) \\ 60.78 \ (\pm 0.24) \\ 64.83 \ (\pm 0.37) \end{array}$

Table 2: Performance on PDTB 3.0 L2 classification.

3.1 Baseline results

Following our proposed protocol, we report baseline results from two strong sentence encoder models: BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019), using a publicly available codebase.⁵ See Appendix C for training details. We present L2 results on PDTB 2.0 in Table 1 and results on PDTB 3.0 in Table 2 (see Appendix D for L1 results). To maintain backwards compatibility to the literature, we also report PDTB 2.0 results on Ji, Lin and P&K splits (see Section 2.1). Ji & Lin are the most common splits, and P&K is the split used by Nie et al. (2019) who claim the current stateof-the-art for L2. For PDTB 2.0 (Table 1), our baselines showed strong performance on all splits. XLNet-large was the single best model, significantly outperforming every best reported result.⁶

3.2 Single-span baselines

Table 4 lists the performance of single-span (either ARG1 or ARG2) baseline models for both PDTB 2.0 and 3.0. This baseline adapts the idea of hypothesis-only baselines in Natural Language Inference (Poliak et al., 2018), where we limit the training data by only showing the models one of the two spans that are in a discourse relation. We discuss these baselines further in Section 4.

4 Discussion: where should we go next?

Annotation improvements in PDTB 3.0 are effective. PDTB 3.0 claims several improvements

³Due to limitations of compute, we only report random restarts of cross-validation (5 seeds x 12 folds) for our main results. For additional experiments in Section 4, we report the average over folds only. Generally, variance over seeds were smaller than over folds for our models.

⁴https://github.com/najoungkim/pdtb3

⁵https://github.com/huggingface/ pytorch-transformers

⁶We used the $N-1 \chi^2$ test to compare proportions instead of a matched test like McNemar's, because we only had access to reported accuracies (rather than raw predictions) of the best models in the literature.

Label	$\mu(train)$	$\mu(test)$	BERT-base	BERT-large	XLNet-base	XLNet-large
Cont.Cause.Reason	2474	238	62.1	64.1	62.8	71.0
Cont.Cause.Result	2378	227	56.1	60.2	60.6	70.6
Expn.Level-of-detail.Arg1-as-detail	214	21	0.0	3.3	7.2	8.0
Expn.Level-of-detail.Arg2-as-detail	2602	240	46.8	52.8	53.2	55.8
Expn.Manner.Arg1-as-manner	480	6	29.6	39.8	49.1	34.8
Expn.Manner.Arg2-as-manner	140	12	49.7	55.3	57.6	57.2
Temp.Asynchronous.Precedence	907	85	59.0	62.3	63.2	68.5
Temp.Asynchronous.Succession	174	16	13.3	31.0	37.1	43.7

Table 3: Average label accuracy per directional label in L2+L3 classification, over cross-validation folds.

Model	X-Accuracy $(\pm \sigma)$
Majority class	25.52
BERT-(base, uncased), ARG1-only	42.28 (±1.76)
BERT-(large, uncased), ARG1-only XLNet-(base, cased), ARG1-only	$\begin{array}{c} 42.79 \ (\pm 1.31) \\ 42.39 \ (\pm 1.03) \end{array}$
XLNet-(large, cased), ARG1-only	42.55 (±1.44)
BERT-(base, uncased), ARG2-only	47.59 (±1.94)
BERT-(large, uncased), ARG2-only	$48.69(\pm 1.57)$
XLNet-(base, cased) ARG2-only	$48.00(\pm 1.97)$
XLNet-(large, cased), ARG2-only	47.99 (±1.72)
BERT-(base, uncased), Upper-bound	61.71 (±0.02)
BERT-(large, uncased), Upper-bound	63.82 (±0.01)
XLNet-(base, cased), Upper-bound	63.43 (±0.01)
XLNet-(large, cased), Upper-bound	63.41 (±0.02)

Table 4: Cross-validation accuracy on PDTB 3.0 L2classification (14-way) of single-span baselines.

over PDTB 2.0. For instance, the annotation manual (Prasad et al., 2019) remarks that LIST was removed since it was "not in practice distinguishable from CONJUNCTION". Indeed, models trained on PDTB 2.0 behaved exactly so, classifying most of LIST as CONJUNCTION (but not vice versa, likely due to frequency effect; see Appendix G). We conducted an additional experiment testing the impact of the new annotation scheme, in an attempt to address the question "If we want to detect relation X in a downstream task, which PDTB should we use to train our models?". We trained the same model (BERT-large) twice on the same set of datapoints, only varying the annotation scheme. Since PDTB 3.0 has both added and removed examples, we filtered the datasets so that the two PDTBs contained exactly the same span pairs. With the model and inputs fixed, the labeling scheme should be the only effective factor. After filtering, the majority-class baseline for both were less than 30%.

Table 5 suggests that PDTB 3.0's annotation scheme does lead to improved distinguishability of CONJUNCTION.⁷ PDTB 3.0 overall yielded better

(or unchanged) distinguishability of shared labels except for CONTRAST. This trend was especially salient for CONCESSION that was practically unlearnable from PDTB 2.0. This supports the utility of PDTB 3.0 over 2.0 if downstream transfer is considered, motivating a transition to 3.0.

Unsurprisingly, the change in distinguishability was highly dependent on the change in label counts in the training data (Table 5, Δ). But change in frequency alone does not give us the full picture. For instance, SYNCHRONOUS remained difficult to learn even with a substantial increase in labeled examples. The absolute size of the class was also not deterministic of performance. There were 192 training instances of SYNCHRONOUS in the filtered PDTB 2.0 and 261 for PDTB 3.0. Similar/smaller classes such as |ALTERNATIVE| = 118 in PDTB 2.0 and |SUBSTITUTION| = 191 in PDTB 3.0 were still learnable with 26% and 48% accuracy, respectively. This was mostly due to SYNCHRONOUS being mislabeled as CONJUNCTION, which was also the case in the unfiltered dataset (see Appendix G).

Label	Acc. (2.0)	Acc. (3.0)	Δ
Cont.Cause	65.3	67.8 *	+25
Comp.Concession	0	46.6***	+740
Comp.Contrast	50.5 *	43.4	-820
Expn.Conjunction	57.6	61.7 **	+88
Expn.Instantiation	60.7	57.7	+4
Temp.Asynchronous	48.8	48.0	-7
Temp.Synchronous	0	2.7	+70

Table 5: Pooled cross-validation accuracy of BERTlarge on shared labels. Models were trained on the same set of datapoints, with only the annotation scheme differing. Δ denotes the average per-fold change in (filtered) training label counts from PDTB 2.0 to 3.0.

New directional labels are potentially useful but distributionally skewed. The new anno-

⁷We used pooled cross-validation accuracy (compared us-

ing Fisher's exact test and Bonferroni correction) because label sparsity made fold-wise comparisons underpowered for small classes like ASYNCHRONOUS.

tation scheme for PDTB 3.0 marks the directionality of relations (e.g., ARG1- vs ARG2-AS-MANNER). These relations are important for naturally-occurring discourse, where order-variable asymmetrical relations are common. For example, in Figure 1, span [2] is conditionally dependent on [3], and [5] has a dependency on [4]; such ordered dependencies must be correctly tracked across discourse contexts. We investigated whether directional labels are sufficiently identifiable with our models. We replaced L2 classes with L3 subclasses (L2+L3), if *both* subclasses had > 100 examples. Except for REASON and RESULT, the distribution of L3 classes under the same L2 is heavily skewed, which led to low performance (Table 3). This calls for a data augmentation that would balance subclass ratios and alleviate label sparsity at L3.

Within-document label distribution is informative, even for shallow discourse parsing. We have advocated for an evaluation scheme that preserves larger contexts. This is motivated by the fact that discourse relations are not independently distributed from one another (even when they are annotated in isolation, as in PDTB). For instance, implicit CONJUNCTION (*IC*) relations are likely to be adjacent; in PDTB 3.0, the probability of one *IC* following another is $P(IC_2|IC_1) = 0.14$, when P(IC) = 0.08. Implicit REASON is likely to be adjacent to RESULT; P(IReason|IResult) =0.12, P(IReason) = 0.05.

Vanilla pretrained encoders are strong, but are overreliant on lexical cues. A simple finetuning of pretrained encoders yielded impressive gains. At the same time, they overrelied on lexical cues. For instance, ARG2-initial *to* often signals PURPOSE; 79.9% of such cases are true PURPOSE relations. It is reasonable for our models to utilize this strong signal, but the association was much amplified in their prediction. For example, XLNetbase predicted PURPOSE for 95.8% of the examples with ARG2-initial *to*. We also found that model predictions were in general brittle; a simplistic lexical perturbation with no semantic effect, such as appending '-' to the beginning of spans, resulted in a 9%p drop in performance for BERT-large models.

Overall, there still remains much overhead for improvement, with our best model at 66% accuracy on PDTB 3.0 L2 classification. Combining pretrained encoders and expanded context modeling to better capture document-level distributional



Figure 1: A snippet of an online document for IT troubleshooting, segmented in discourse units.

signals could be a promising next step.

Aggregation of single-span baselines as decontextualized upper-bounds. Lexical cues continue to be informative even for implicit relations, as with the case of ARG2-initial to. Although these signals could be genuine rather than artifactual, they require comparatively less multi-span reasoning. Then, how much of our dataset only requires shallower reasoning as such? To address this question, we constructed a decontextualized baseline by aggregating predictions of single-span models, and assuming that an oracle always chooses the right answer if it is in the prediction set. This provides an upper-bound estimate of the performance of a model that only disjointly considers the two input spans, but still has full lexical access. Comparing the final rows of Table 4 and Table 2, we see that no model reliably outperforms its decontextualized upper-bound counterpart.

5 Conclusion

We have surveyed the literature to highlight experimental inconsistencies in implicit discourse relation classification, and suggested an improved protocol using section-level cross-validation. We provided a set of strong baselines for PDTB 2.0 and 3.0 following this protocol, as well as results on a range of existing setups to maintain comparability. We discussed several future directions, including data augmentation for downstream transferability, applicability of pretrained encoders to discourse, and utilizing larger discourse contexts.

Acknowledgments

This work was supported by IBM Research. We thank the three anonymous reviewers for their insightful comments. We also thank Sadhwi Srinivas, Grusha Prasad and Tal Linzen for their advice on statistical analysis.

References

- Hongxiao Bai and Hai Zhao. 2018. Deep enhanced representation for implicit discourse relation recognition. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 571– 583, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Hongxiao Bai, Hai Zhao, and Junhan Zhao. 2019. Memorizing all for implicit discourse relation recognition. arXiv:1908.11317v1.
- Chloé Braud and Pascal Denis. 2015. Comparing word representations for implicit discourse relation classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2201–2211, Lisbon, Portugal. Association for Computational Linguistics.
- Jifan Chen, Qi Zhang, Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Implicit discourse relation detection via a deep architecture with gated relevance network. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1726– 1735, Berlin, Germany. Association for Computational Linguistics.
- Zeyu Dai and Ruihong Huang. 2018. Improving implicit discourse relation classification by modeling inter-dependencies of discourse units in a paragraph. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 141–151, New Orleans, Louisiana. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Song Feng, Kshitij Fadnis, Q Vera Liao, and Luis A Lastras. 2020. Doc2Dial: a framework for dialogue composition grounded in documents. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)*.
- Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the* 57th Annual Meeting of the Association for Computational Linguistics, pages 2786–2791, Florence, Italy. Association for Computational Linguistics.
- Peter Jansen, Mihai Surdeanu, and Peter Clark. 2014. Discourse complements lexical semantics for nonfactoid answer reranking. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),

pages 977–986, Baltimore, Maryland. Association for Computational Linguistics.

- Yangfeng Ji and Jacob Eisenstein. 2015. One vector is not enough: Entity-augmented distributed semantics for discourse relations. *Transactions of the Association for Computational Linguistics*, 3:329–344.
- Man Lan, Jianxiang Wang, Yuanbin Wu, Zheng-Yu Niu, and Haifeng Wang. 2017. Multi-task attentionbased neural networks for implicit discourse relationship representation and identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1299– 1308, Copenhagen, Denmark. Association for Computational Linguistics.
- Wenqiang Lei, Yuanxin Xiang, Yuwei Wang, Qian Zhong, Meichun Liu, and Min-Yen Kan. 2018. Linguistic properties matter for implicit discourse relation recognition: Combining semantic interaction, topic continuity and attribution. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Junyi Jessy Li and Ani Nenkova. 2014. Reducing sparsity improves the recognition of implicit discourse relations. In Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), pages 199–207, Philadelphia, PA, U.S.A. Association for Computational Linguistics.
- Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. 2009. Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 343–351, Singapore. Association for Computational Linguistics.
- Yang Liu and Sujian Li. 2016. Recognizing implicit discourse relations via repeated reading: Neural networks with multi-level attention. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1224–1233, Austin, Texas. Association for Computational Linguistics.
- Annie Louis, Aravind Joshi, Rashmi Prasad, and Ani Nenkova. 2010. Using entity features to classify implicit discourse relations. In *Proceedings of the SIG-DIAL 2010 Conference*, pages 59–62, Tokyo, Japan. Association for Computational Linguistics.
- Linh The Nguyen, Linh Van Ngo, Khoat Than, and Thien Huu Nguyen. 2019. Employing the correspondence of relations and connectives to identify implicit discourse relations via label embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4201– 4207, Florence, Italy. Association for Computational Linguistics.
- Allen Nie, Erin Bennett, and Noah Goodman. 2019. DisSent: Learning sentence representations from explicit discourse relations. In *Proceedings of the*

57th Annual Meeting of the Association for Computational Linguistics, pages 4497–4510, Florence, Italy. Association for Computational Linguistics.

- Joonsuk Park and Claire Cardie. 2012. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 108–112, Seoul, South Korea. Association for Computational Linguistics.
- Gary Patterson and Andrew Kehler. 2013. Predicting the presence of discourse connectives. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 914–923, Seattle, Washington, USA. Association for Computational Linguistics.
- Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. 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*, pages 683–691, Suntec, Singapore. Association for Computational Linguistics.
- Emily Pitler, Mridhula Raghupathy, Hena Mehta, Ani Nenkova, Alan Lee, and Aravind Joshi. 2008. Easily identifiable discourse relations. In *COLING 2008: Companion volume: Posters*, pages 87–90, Manchester, UK. COLING 2008 Organizing Committee.
- Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics* (*SEM), pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.
- Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *LREC 2008*.
- Rashmi Prasad, Bonnie Webber, Alan Lee, and Aravind Joshi. 2019. Penn Discourse Treebank Version 3.0. In *LDC2019T05*. Philadelphia: Linguistic Data Consortium.
- Lianhui Qin, Zhisong Zhang, Hai Zhao, Zhiting Hu, and Eric Xing. 2017. Adversarial connectiveexploiting networks for implicit discourse relation classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1006–1017, Vancouver, Canada. Association for Computational Linguistics.
- Attapol Rutherford, Vera Demberg, and Nianwen Xue. 2017. A systematic study of neural discourse models for implicit discourse relation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume*

1, Long Papers, pages 281–291, Valencia, Spain. Association for Computational Linguistics.

- Marzieh Saeidi, Max Bartolo, Patrick Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2087–2097, Brussels, Belgium. Association for Computational Linguistics.
- Wei Shi and Vera Demberg. 2017. Do we need cross validation for discourse relation classification? In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 150–156, Valencia, Spain. Association for Computational Linguistics.
- Wei Shi and Vera Demberg. 2019. Learning to explicitate connectives with Seq2Seq network for implicit discourse relation classification. In Proceedings of the 13th International Conference on Computational Semantics - Long Papers, pages 188–199, Gothenburg, Sweden. Association for Computational Linguistics.
- WenTing Wang, Jian Su, and Chew Lim Tan. 2010. Kernel based discourse relation recognition with temporal ordering information. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 710–719, Uppsala, Sweden. Association for Computational Linguistics.
- 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*, pages 1–16, Beijing, China. Association for Computational Linguistics.
- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 5753–5763. Curran Associates, Inc.

Appendix

A Dataset Statistics

We report the training, development and test set sizes for all dataset splits discussed in the paper (Table 6). These are counts of individual labeled span pairs in the dataset, not the counts of individual labels (development and test set examples can be doubly-annotated). Note that the count we provide for the training split of Ji is one short of what has been reported in Shi and Demberg (2019) and also the count obtained by using Qin et al. (2017)'s preprocessing code. This is due to a duplicate example with label EXPANSION.ALTERNATIVE, which our preprocessing code does not generate.

Split	Train	Dev	Test		
	PDTB 2.0				
Ji	12825	1165	1039		
Lin	13366	515	766		
P&K	13908	1165	1188		
X-val	13676	1281	1273		
L1 (Ji)	13046	1183	1046		
	PDTB 3.0				
L2 X-val	19005	1756	1747		
L2+L3 X-val	19005	1756	1747		
L1 (Ji)	17854	1647	1471		

Table 6: Dataset sizes for PDTB 2.0 and 3.0. Cross-validation counts are averaged across 12 folds.

Tables 7	7 and 8	list the	label	counts	of	each	ı cl	lass in
PDTB 3	3.0 and	PDTB	2.0, r	respectiv	vel	y.		

Label	n
Comparison	2298/2518
Contingency	6998/7583
Expansion	10062/10833
Temporal	1731/1828
Comparison.Concession	1494
Comparison.Contrast	983
Contingency.Cause	5785
Contingency.Cause+Belief	202
Contingency.Condition	199
Contingency.Purpose	1373
Expansion.Conjunction	4386
Expansion.Equivalence	336
Expansion.Instantiation	1533
Expansion.Level-of-detail	3361
Expansion.Manner	739
Expansion.Substitution	450
Temporal.Asynchronous	1289
Temporal.Synchronous	539
Contingency.Cause.Result	2835
Contingency.Cause.Reason	2950
Expansion.Level-of-detail.Arg1-as-detail	256
Expansion.Level-of-detail.Arg2-as-detail	3105
Expansion.Manner.Arg1-as-manner	572
Expansion.Manner.Arg2-as-manner	167
Temporal.Asynchronous.Precedence	1081
Temporal.Asynchronous.Succession	208

Table 7: Label counts for PDTB 3.0 L1, L2 and directional senses of L3 that have more than 100 annotated instances. L1 classification is evaluated on Ji split, so we list both the label counts in Ji split and the total label counts in the whole dataset.

Label	п
Comparison	2291/2503
Contingency	3911/4255
Expansion	8249/8861
Temporal	909/950
Comparison.Concession	223
Comparison.Contrast	2120
Contingency.Cause	4172
Contingency.Pragmatic cause	83
Expansion.Conjunction	3534
Expansion.Instantiation	1445
Expansion.Alternative	185
Expansion.List	400
Expansion.Restatement	3206
Temporal.Asynchronous	697
Temporal.Synchrony	251

Table 8: Label counts for PDTB 2.0 L1 and 11 senses of L2 (label set commonly used in the literature for L2 classification). L1 classification is evaluated on Ji split, so we list both the label counts in Ji split and the total label counts in the whole dataset.

B List of Splits in Prior Work

We compile a (non-exhaustive) list of the Wall Street Journal sections used as training, development, test sets in published work to demonstrate the high variability. We mostly list works that do not explicitly specify the source of the splits, with some exceptions. Some of the works have overlapping sections across splits, which we suspect to be typos but cannot verify.

- Prasad et al. (2008) (officially recommended split): 2-21 (train), 22 (dev), 23 (test)
- Pitler et al. (2009); Ji and Eisenstein (2015): 2-20 (train), 0-1 (dev), 21-22 (test)
- Lin et al. (2009): 2-21 (train), 23 (test)
- Patterson and Kehler (2013): 2-22 (train), 0-1 (dev), 23-24 (test)
- Wang et al. (2010): 2-22 (train), 23-24 (test)
- Louis et al. (2010): 0-22 (train), 23-24 (test)
- Braud and Denis (2015): 2-21 (train), 0-1, 23-24 (dev), 21-22 (test)
- Li and Nenkova (2014): 2-19 (train), 20-24 (test)
- Lei et al. (2018): 2-20 (train), 0-1, 23-24 (dev), 21-22 (test)
- Park and Cardie (2012): 2-20 (train), 0-2 (dev), 21-22 (test)

C Training Details

For all sentence encoder models, we fine-tuned each encoder for a maximum of 10 epochs with early stopping when the the development set performance did not improve for 5 evaluation steps (step size=500), with a batch size of 8. We used a learning rate of 5e-6 for all models except for XLNet-large, for which we used 2e-6. We used accuracy as the validation metric. We ran each model 5 times with different random initializations of the fine-tuning layer, and reported the average performance across the 5 runs.

D Top-level Sense Classification Results

Table 9 shows the performance on L1 classification for both PDTB 2.0 and PDTB 3.0.

Model	PDT	B 2.0	PDT	B 3.0
	F1	Acc	F1	Acc
Majority class	17.4	54.9	15.2	47.3
Lan et al. (2017)	47.8	57.4	-	-
Dai and Huang (2018)	48.7	58.2	-	-
Bai and Zhao (2018)	51.1	-	-	-
Bai et al. (2019)	52.2	60.7	-	-
Nguyen et al. (2019)	53.0	-	-	-
BERT (base, uncased)	52.6	64.3	62.1	69.0
BERT (large, uncased)	59.1	68.7	66.8	72.4
XLNet (base, cased)	56.0	66.3	64.8	71.3
XLNet (large, cased)	54.3	67.2	68.3	73.8

Table 9: Accuracy and F1 on L1 classification (4-way) for PDTB 2.0 and 3.0, using Ji split for both. We report average performance across 5 random restarts.

E Single-span Baselines for L2 Classification

Table 10 lists the performance of single-span (either ARG1 or ARG2) baselines for PDTB 2.0. Results on PDTB 3.0 are reported in Table 4.

We additionally note that ARG2-only models consistently outperform ARG1-only models in both PDTB 2.0 and 3.0. For PDTB 3.0, the strong association between ARG2-initial *to* and CONTIN-GENCY.PURPOSE was largely responsible for this discrepancy (see Section 4 also).

F Cross-validation and Randomized validation

Gorman and Bedrick (2019) have proposed validation over randomized splits using significance testing with multiple-comparisons correction. An adaptation of this idea to our proposal of sectionbased evaluation would be randomized sampling of sections to create section-based splits. Given label sparsity and distributional skew across sections, cross-validation has an advantage of guaranteed coverage of label counts used for testing, although this may not be a large issue if sufficient number of random splits are sampled. Conversely, the main goal of evaluation on random splits-avoiding overfitting to the standard split—is partially mitigated by reporting the average performance over crossvalidation splits. Still, if a standard cross-validation split is adopted, overfitting may still arise over time. Although we leave it to future work to decide which practice should be followed, we provide comparisons between the four models we tested, using our proposed cross-validation splits and random validation splits (both n = 12). Random splitting was done section-wise instead of instance-wise; we randomly split the dataset into 21 train, 2 dev, 2 test sections 12 times. Table 11 shows the model comparison results.

G Additional Error Analyses

Figure 2 shows the confusion matrices generated from PDTB 2.0 L2 classification results produced by XLNet-large and BERT-large models. Figure 3 shows the confusion matrices of PDTB 3.0 L2 classification predictions, again from XLNet-large and BERT-large models (we did not observe immediate qualitative differences between XLNet and BERT, or between large and base models).

The figures aggregate the predictions from all test sets of the cross-validation experiment, so the datapoints shown span the full dataset except for WSJ section 22. The colors are normalized over each row; the darkest shade is the most frequently predicted label for the true label denoted by the row.

It was generally the case for both models that classes sharing the same L1 senses (e.g., CONTIN-GENCY.CAUSE and CONTINGENCY.PRAGMATIC CAUSE, or COMPARISON.CONTRAST and COM-PARISON.CONCESSION) were confused. When such confusions occurred, the more frequent class often subsumed the prediction of the other class (e.g., CONTINGENCY.PRAGMATIC CAUSE was often classified as CONTINGENCY.CAUSE but not vice versa).

As noted in Section 4, TEMPO-RAL.SYNCHRONOUS (SYNCHRONY in PDTB

		Accuracy		X-Accuracy
Model	Ji	Lin	P&K	
Majority class	26.18	26.11	28.54	26.42
Adversarial Net (Qin et al., 2017)	46.23	44.65	-	-
Seq2Seq+MemNet (Shi and Demberg, 2019)	47.83	45.82	-	41.29
ELMo (Bai and Zhao, 2018)	48.22	45.73	-	-
ELMo, Memory augmented (Bai et al., 2019)	49.15	46.08	-	-
Multitask learning (Nguyen et al., 2019)	49.95	46.48	-	-
BERT+MNLI (Nie et al., 2019)	-	-	53.7	-
BERT+DisSent Books 5 (Nie et al., 2019)	-	-	54.7	-
BERT (base, uncased), ARG1-only	38.59 (±0.67)	36.11 (±1.01)	35.86 (±1.43)	36.66 (±1.26)
BERT (large, uncased), ARG1-only	39.31 (±0.70)	36.42 (±0.21)	37.71 (±1.42)	37.23 (±1.22)
XLNet (base, cased), ARG1-only	39.48 (±1.10)	35.40 (±1.06)	35.71 (±1.32)	37.38 (±1.76)
XLNet (large, cased), ARG1-only	39.77 (±1.58)	35.61 (±1.48)	$36.20(\pm 1.77)$	36.33 (±2.04)
BERT (base, uncased), ARG2-only	40.99 (±1.34)	40.99 (±1.34)	40.98 (±1.12)	40.60 (±1.48)
BERT (large, uncased), ARG2-only	44.27 (±1.00)	40.78 (±1.33)	42.34 (±1.21)	41.45 (±1.64)
XLNet (base, cased), ARG2-only	43.20 (±1.48)	40.84 (±0.99)	40.45 (±1.22)	$40.46(\pm 1.45)$
XLNet (large, cased), ARG2-only	42.00 (±1.24)	41.78 (±1.00)	41.48 (±1.14)	41.17 (±1.48)

Table 10: Single-span baseline performance on PDTB 2.0 L2 classification (11-way). All results are averages over 5 random restarts, except for cross-validation where we report averages over 12 folds.

	X-validation	Randomized
BERT-base vs BERT-large	8	9
BERT-base vs XLNet-base	8	6
BERT-base vs XLNet-large	12	12
BERT-large vs XLNet-large	6	7
XLNet-base vs BERT-large	0	1
XLNet-base vs XLNet-large	6	10

Table 11: The number of splits out of twelve for which the second model had significantly higher accuracy than the first model after Bonferroni correction. We used McNemar's test following Gorman and Bedrick (2019).

2.0) was frequently confused with EXPAN-SION.CONJUNCTION (but not vice versa). The models generally had a tendency to predict CONTINGENCY.CAUSE across the board, likely due to it being the most frequent label.



Figure 2: Confusion matrices of XLNet-large and BERT-large models on PDTB 2.0 L2 classification task. The rows are true labels and the columns are predicted labels.



Figure 3: Confusion matrices of XLNet-large and BERT-large models on PDTB 3.0 L2 classification task. The rows are true labels and the columns are predicted labels.