Contextualized Embeddings for Connective Disambiguation in Shallow Discourse Parsing

René Knaebel
Applied Computational Linguistics
Department of Linguistics
University of Potsdam
Germany
rknaebel@uni-potsdam.de

Manfred Stede
Applied Computational Linguistics
Department of Linguistics
University of Potsdam
Germany
stede@uni-potsdam.de

Abstract

This paper studies a novel model that simplifies the disambiguation of connectives for explicit discourse relations. We use a neural approach that integrates contextualized word embeddings and predicts whether a connective candidate is part of a discourse relation or not. We study the influence of those context specific-embeddings. Further, we show the benefit of training the tasks of connective disambiguation and sense classification together at the same time. The success of our approach is supported by state-of-the-art results.

1 Introduction

Coherence is crucial for humans to be able to interpret text. The area of discourse parsing models this by identifying certain phrases (arguments) within a text and using discourse relations to unfold their underlying connections. These discourse relations and their understanding are important for tasks such as machine translation (Sim Smith, 2017), abstractive summarization (Wu and Hu, 2018), and text simplification (Zhong et al., 2020). 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*. However, such cues can be ambiguous, as they may signal more than one relation type or may not always function as a relation indicator. Two challenges arise[1]—first, distinguishing connectives from words with mere sentential meaning:

1. Mr. Perkins believes, **however**, that the market could be stabilized.

2. “The 1987 crash was a false alarm **however** you view it,” says university of Chicago economist.

Here, example 1 shows a discourse relation, while example 2 uses ‘however’ in its sentential reading. The second challenge consists in classifying a connective’s sense (described in detail in Section 2):

3. She owns a bike, **while** her brother drives a car. (Comparison.Contrast)

4. You should take the deal **or** even try to negotiate this price down. (Expansion.Alternative)

5. **If** things work out, **then** everybody will be happy. (Contingency.Condition)

6. **While** it is raining outside, I clean the dishes. (Temporal.Synchronous)

Shallow discourse parsing (SDP) is the area that builds models to uncover such discourse structures 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 (which hold between adjacent text spans without a lexical signal being present). In this work, we focus in particular on the binary *connective disambiguation* of explicit discourse relations and, further, integrate explicit *sense prediction* into our model, as those two tasks are highly related.

Word embeddings provide dense token representations in a low-dimensional vector space pretrained on large unannotated text corpora. First, we use fastText (Bojanowski et al., 2017), which is based on the skip-gram model (Mikolov et al., 2013) and integrates character n-grams into its representation. Second, we use GloVe (Pennington et al., 2014); as opposed to fastText, those embeddings were calculated through co-occurrence statistics rather than trained by a neural network. Recently, models were introduced that provide *contextualized* word embeddings (Peters et al., 2018; Devlin et al., 2019) on

[1] Examples 1–2 are from PDTB (see Section 2); examples 3–6 are artificially constructed; senses follow PDTB2 style.

Proceedings of the First Workshop on Computational Approaches to Discourse, pages 65–75
Online, November 20, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

demand and thus tackle the problem of identical representations for homonymous words with different senses, which had been indistinguishable in older models. For our experiments, we use BERT (Devlin et al., 2019), which was successful in many areas of NLP (Liu and Lapata, 2019; Liu et al., 2019).

In this work, we present a novel approach to identifying explicit relations in shallow discourse parsing. We introduce a simple yet powerful model that outperforms previous research on the binary disambiguation of connective candidates. Furthermore, we adopt connective sense classification as an auxiliary task to improve performance and generalization capabilities and study the benefits of jointly training the auxiliary task in addition to the main task. This is because, in various cases, training neural models on multiple related tasks has shown beneficial for the learned representation (Caruana, 1993), as it introduces inductive bias and, thereby, reduces the possible hypothesis space (Baxter, 2000). Specifically, the work of Collobert et al. (2011) has pointed out the advantages of multitask learning on NLP tasks. We compare our results with state-of-the-art SDP components that took part at the CoNLL Shared Task in 2016.[2] The contributions of this paper are as follows:

1. We design a simple neural architecture that eliminates the need for hand-engineered features. To the best of our knowledge, this work is the first to provide state-of-the-art performance on word-embedding-based connective disambiguation.

2. We present a novel approach that successfully combines the two tasks of connective disambiguation and explicit sense classification into one single model. In contrast to previous work, we introduce a more sensitive measure and, with its help, demonstrate improved stability of the jointly trained model.

In the following, Section 2 describes the corpus for the experiments; Section 3 explains our method. The experiments and results are presented in Section 4 and Section 5; Section 6 discusses relevant related work, followed by conclusions in Section 7.

2 Penn Discourse Treebank

Shallow discourse parsing is a challenging task that was promoted by the development of the second version of the Penn Discourse Treebank (PDTB2) (Prasad et al., 2008). This corpus provides about 43,000 annotated discourse relations, of which roughly 18,000 are signalled by explicit discourse connectives. Those relations are further annotated with a three-level sense hierarchy (one or two senses per relation). All discourse relations consist of two arguments and are associated with one of various types; the focus of our work is on relations of the *explicit* type.

Coarse Class	**Absolute**	**Relative**
NoConn	34174	69.90
Expansion	5007	10.24
Comparison	4382	8.96
Temporal	2752	5.63
Contingency	2578	5.27
Fine Class		
NoConn	34174	70.44
Expansion.Conjunction	4323	8.91
Comparison.Contrast	2956	6.09
Contingency.Condition	1147	2.36
Temporal.Sync	1133	2.34
Comparison.Concession	1079	2.22
Contingency.Cause.Reason	943	1.94
Temporal.Async.Succession	842	1.74
Temporal.Async.Precedence	770	1.59
Contingency.Cause.Result	487	1.00
Expansion.Instantiation	236	0.49
Expansion.Alternative	195	0.40
Expansion.Restatement	121	0.25
Expansion.Alternative.Chosen	95	0.20
Expansion.Exception	13	0.03

Table 1: Class distribution of the training data.

The Shared Tasks at CoNLL 2015 and 2016 (Xue et al., 2015, 2016) used PDTB2 with minor changes. Successful systems were Wang et al. (2015); Wang and Lan (2016); Oepen et al. (2016). They largely follow a pipeline architecture (Lin et al., 2014), which consists of successive tasks of connective identification, argument labeling, and sense classification for both explicit and implicit relations.

Recently, PDTB3 (Prasad et al., 2018) was published, which extends the previous work with more available relations and corrects several former annotations. The authors also adjusted the relations' sense labels for a more balanced class distribution. For the sake of comparison with previous work on SDP, we stick to the PDTB2 corpus and assume to achieve similar results with PDTB3.

Table 1 summarizes the distribution regarding sense classes, where we denote candidate words with sentential reading by NoConn. In both settings, NoConn dominates other classes and, thus, serves

[2] We are not aware of more recent results.

as the majority baseline. The first setting shows the four *coarse* sense classes provided by PDTB2. The second setting describes the *fine* senses as defined in the Shared Task. In contrast to the first setting, the distribution slightly changes, as rare training samples were removed or combined with other classes. Although the exact numbers for NoConn are the same in both settings, the ratios are different, which can be explained by the small modifications made to PDTB2 in the competitions.

3 Method

This work introduces a first, simple neural architecture for shallow discourse connective disambiguation. The system builds upon previous observations that a word's context could be used as a strong indicator for the presence of a discourse relation (Lin et al., 2014).[3] Our work investigates the limitations of knowledge free approaches and introduces a simple yet flexible model without domain knowledge.

We assume that word embeddings contain information about the discourse that can be used for the disambiguation task. We study standard noncontextualized embeddings (in particular, GloVe embeddings and Wikipedia-based fastText embeddings) and compare those to the recently developed contextualized embeddings (represented by BERT). We first hypothesize that contextualized embeddings yield better results than their noncontextualized counterpart. Second, we expect the context span to influence the model's performance, as the context may indicate a word's function more clearly.

In addition, we propose a second model based on the first one, which successfully combines connective disambiguation with sense classification as an auxiliary task. We follow the idea of previous work that sense classification can be performed without extracting the connectives' arguments (Pitler and Nenkova, 2009; Lin et al., 2014; Qin et al., 2016). Further, it has been previously shown that, for the identification of an explicit relation's sense, the connective itself as well as its context already provide significant information (Pitler and Nenkova, 2009; Lin et al., 2014; Wang and Lan, 2015; Ghosh et al., 2011). Consequently, we assume the necessary information for sense classification to be already accessible by our neural connective disambiguation model to some degree. Also, this approach eliminates the error propagation and the performance of our joint model stays as is without relying on previous predictions. The reason for adding sense classification as an auxiliary task in the first place is that joint training with auxiliary tasks has shown benefits in earlier work, as mentioned in Section 1. We could validate that this is the case with our connective disambiguation task as well, as later demonstrated in our experiments (see Section 4.2).

In the following sections, we explain in more detail our binary connective disambiguation approach and the joint sense classification model.

3.1 Embedding-Based Connective Disambiguation

For parsing explicit discourse relations, the first task usually involves the identification of possible connective *candidates*. For this purpose, we use a list of candidate patterns based on PDTB2. Some candidates might look like discourse connectives, however, they might only be in sentential use.

Connective annotation in PDTB2 is quite flexible. Connectives can be individual words ('indeed'), multiple consecutive words ('in the end'), or distant words that function together ('neither … nor'). In addition, they can contain adverbial modifications ('at least when,' 'even when,' 'usually when'), which vastly increases the number of possible connectives. Regarding this problem, the CoNLL Shared Task introduced a mapping that normalizes instances of connectives to their *head* by removing adverbial modifiers. For example, the three full connectives above all normalize to their head 'when.' For our studies, we follow this approach and focus on the disambiguation of connective head candidates rather than fully annotated connectives as in the original corpus.

We introduce a simple neural architecture (see Figure 1) that relies on pretrained word embeddings instead of hand-engineered features. The network consists of a multilayer perceptron with a single hidden layer. As the network's input, we use the candidate word's embeddings and its context.

A continuous token sequence of length n is encoded as an embedding sequence $(e_1, e_2, \ldots, e_n)$. We define our input with regards to the candidate's positions within the sentence (denoted as C) and use *cmin* and *cmax* for the first and last occurrence of the candidate, respectively. Finally, with a con-

[3] Note that a word's *context* refers to the words immediately preceding and succeeding it, which is not to be confused with contextualized or noncontextualized word *embeddings*.

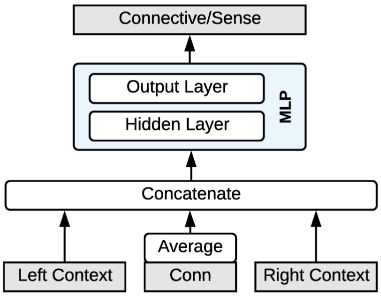


Figure 1: Model overview. The average of the connective embeddings and their context serve as input, a single hidden layer is used for transformation, and the final layer outputs either the connective probability or sense classes.

text size of s, our input looks as follows:

$$\begin{aligned} ctx_{left} &= (e_{cmin-s}, \dots, e_{cmin-1}) \\ conn &= (e_c : c \in C) \\ ctx_{right} &= (e_{cmax+1}, \dots, e_{cmax+s}) \end{aligned}$$

Because the candidate might consist of multiple words ('in particular'), we simply average all candidate embeddings and concatenate remaining embeddings to build the network's input x:

$$x = ctx_{left} + \overline{conn} + ctx_{right}$$

Thus, independent of the number of words describing a connective candidate, the input always has the same dimension. We do not average the embeddings of the context because this would lead to unwanted information loss.

For the transformation of candidates and their context into word embeddings, we use the tokenization provided by the CoNLL Shared Tasks. No other annotations such as POS, constituent trees, and dependencies are used for our experiments. GloVe and fastText are used for noncontextualized embeddings and BERT for contextualized embeddings. For contextualized embeddings, we noticed a difference in the tokenization of contractions. Therefore, we simply replaced occurrences of the token 'n't' by 'not' without changing the overall meaning.

The usage of embeddings is straightforward. Each token in a document is mapped to its embedding representation. In contrast, the contextualized embeddings are generated sentence-wise before extracting context and candidate embeddings. The input for BERT is prepared with special tags for sentence beginnings and ends. Further, original tokens might be split into smaller tokens based on the WordPiece tokenizer (Wu et al., 2016) before feeding them into BERT's encoder. This possibly leads to a higher number of BERT subtoken embeddings than tokens defined on the original corpus. Based on the alignment of original tokens and BERT subtokens, only the first BERT subtoken embedding of a corresponding token is used as its embedding. This selection follows the original BERT publication (Devlin et al., 2019), where features were extracted and the finally predicted classes only rely on the first subtoken's position.

3.2 Joint Disambiguation and Sense Classification

For the reasons explained in the beginning of Section 3, we combine binary connective disambiguation and sense classification into a single, second model. Thus, the model jointly learns whether a connective candidate serves as a discourse signal and, if so, determines its sense. We use the same model as in our previous experiment (see Figure 1) but introduce a novel prediction scheme for the joint classification. As both tasks have exclusive classes, our model either predicts whether a candidate is without sense, which is equivalent to having sentential reading, or predicts one of the desired sense classes. Combining multiple tasks into a single model is called multitask learning (see Section 6).

4 Evaluation

For our experiments, we use PDTB2 (Prasad et al., 2008), especially the version provided for the CoNLL Shared Task. We distinguish between *coarse* senses, which come from the original PDTB, and *fine* senses as defined by the Shared Task. Also, an official split is provided that makes comparisons to other systems more reliable. In particular, this means that we used folders 02–22 for training, folders 00 and 01 as a development set, and folders 23 and 24 for testing. We downloaded word embeddings for GloVe[4] and fastText[5] from their corresponding websites. For the contextualized embeddings, we extracted token embeddings that we had previously transformed using BERT.[6]

As we work with highly imbalanced data, we present our results using precision, recall, and F1 score. Typically, there is a natural inverse rela-

[4] `nlp.stanford.edu/data/glove.6B.zip`

[5] `dl.fbaipublicfiles.com/fasttext/vectors-english/wiki-news-300d-1M.vec.zip`

[6] We use the `bert-base-uncased` model provided by Huggingface's transformer (Wolf et al., 2019).

Model	Conn Disambiguation	Coarse Sense Classification		Fine Sense Classification	
	$F1_{conn}$	$F1_{conn}$	$F1_{sense}$	$F1_{conn}$	$F1_{sense}$
Standard WSJ Test (Section 23)					
bert-ctx-1	**97.32** (99.20)	96.98 (99.77)	**93.03**	96.63 (99.78)	84.17
bert-ctx-0	97.20 (99.29)	**97.45** (99.81)	92.12	**97.18** (99.76)	**86.26**
bert-ctx-2	96.97 (99.08)	95.96 (99.71)	89.57	95.81 (99.72)	81.94
baseline	95.46 (97.11)	—	—	—	—
fasttext-ctx-1	92.09 (94.95)	92.26 (98.61)	87.51	92.63 (98.45)	80.39
glove-ctx-2	92.02 (94.58)	92.03 (98.52)	85.56	90.13 (98.20)	82.35
glove-ctx-1	91.76 (94.69)	91.90 (98.44)	87.22	91.30 (98.18)	78.17
fasttext-ctx-2	91.29 (94.98)	92.51 (98.86)	88.66	91.77 (98.47)	78.18
glove-ctx-0	84.99 (80.98)	84.33 (92.36)	75.53	84.80 (92.62)	66.19
fasttext-ctx-0	84.79 (80.72)	84.20 (92.27)	77.62	84.23 (92.59)	68.79
Wikipedia Blind Test					
bert-ctx-0	**97.03** (98.52)	**96.75** (99.74)	88.79	96.28 (99.72)	**71.69**
bert-ctx-1	96.98 (98.38)	96.18 (99.65)	**90.41**	**96.31** (99.69)	71.51
bert-ctx-2	96.40 (97.29)	95.09 (99.56)	87.92	94.01 (99.54)	66.97
baseline	94.50 (95.06)	—	—	—	—
fasttext-ctx-2	88.99 (90.31)	89.34 (97.95)	82.05	88.26 (97.42)	59.81
fasttext-ctx-1	88.74 (90.10)	88.01 (97.56)	78.46	89.74 (98.06)	62.74
glove-ctx-1	87.86 (88.80)	87.39 (96.95)	77.85	87.78 (97.19)	62.92
glove-ctx-2	87.54 (87.61)	87.63 (97.45)	78.31	87.01 (97.10)	61.15
glove-ctx-0	81.86 (73.77)	82.36 (90.66)	64.16	81.87 (90.91)	40.46
fasttext-ctx-0	81.76 (73.64)	82.98 (90.70)	67.61	82.28 (90.87)	45.68

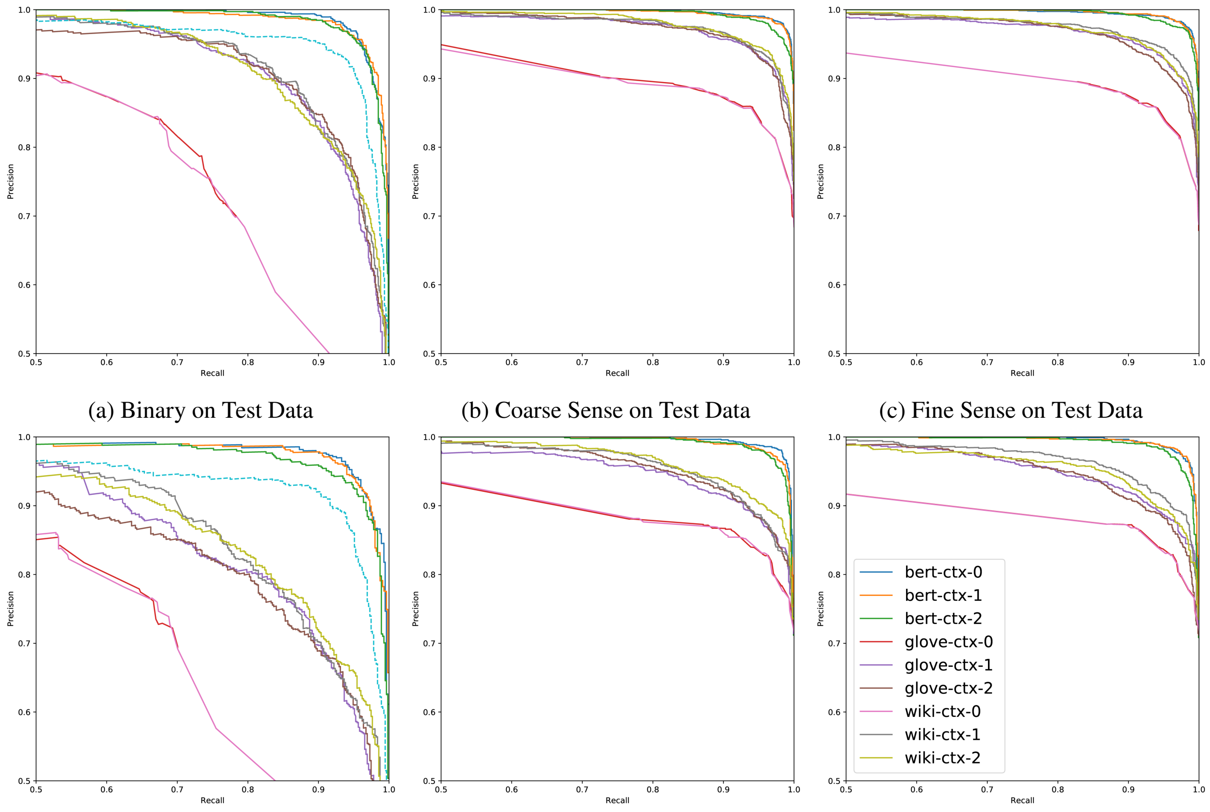
Table 2: Experimental results for various embedding types (GloVe, fastText, BERT) and context sizes (*ctx*). Evaluation involves Section 23 of WSJ and the blind data set proposed for CoNLL Shared Task. All tasks are measured using F1 scores. Average precision is calculated for connective disambiguation and shown in parentheses. Results are ordered by primary task and separated with regards to the groups highlighted in Figure 2.

tion between precision and recall—as one increases, the other decreases. Depending on the final usage, either one could be optimized. While in previous work, scores were usually reported for one specific threshold only, we decided to use precision–recall curves for our experimental results. These give a better understanding of the models' sensitivity to the selected threshold in our binary disambiguation task. To approximate the area under the precision–recall curve, we compute the average precision (AP) score. While F1 score indicates performance for a single threshold only, the AP score helps to compare the precision–recall curves of various models.

In our experiments, we study different embedding types (GloVe, fastText, BERT) with a varying context size ($ctx \in \{0, 1, 2\}$). The dimension (*emb*) of the noncontextualized word embeddings is 300 and 768 for contextualized embeddings, which results in an input size of $(2 * ctx + 1) * emb$. The size of the hidden layer is 2048. All models were trained for at most 50 epochs using early stopping (Prechelt, 1998) when validation loss did not improve over 10 epochs, a batch size of 128, and the Adam (Kingma and Ba, 2015) optimizer with a learning rate of 0.001. For comparison, we also provide a baseline from a reimplementation of Lin et al. (2014). Table 2 reports the performances of our models per experimental setting for each data partition (test and blind) and highlights best performances. In the remainder of this section, we discuss the experimental results obtained for the models presented in Sections 3.1 and 3.2.

4.1 Embedding-Based Connective Disambiguation

Table 2 shows that for contextualized word embeddings, our model generally outperforms the baseline, in particular, the `bert-ctx-1` configuration. This confirms our hypothesis that it is possible to disambiguate connectives to a good extent by using word embeddings. Further, certain groups of experiments can be visually distinguished (see Figures 2a). The weakest performance was achieved when using noncontextualized word embeddings and zero context. This is probably due to the ambiguity of words that have multiple senses. Models that take context into account clearly outperform those with standalone embeddings. From this, we



(a) Binary on Test Data

(b) Coarse Sense on Test Data

(c) Fine Sense on Test Data

(d) Binary on Blind Data

(e) Coarse Sense on Blind Data

(f) Fine Sense on Blind Data

Figure 2: Precision–recall curves for connective disambiguation for the different models—binary disambiguation (left), joint coarse sense (middle), joint fine sense (right)—and data sets—test (top) and blind (bottom). The baseline is shown as a dashed line.

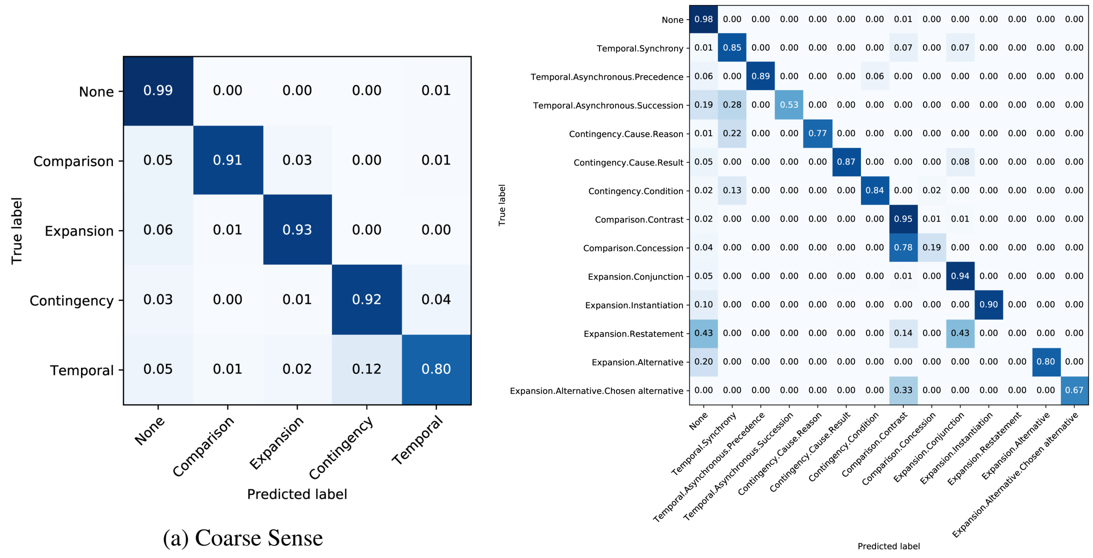
conclude that single noncontextualized word embeddings do not contain enough discourse information but that a model can compensate the missing information with the connective's context. Finally, contextualized embeddings seem to already contain this discourse information, as varying context sizes did not lead to clearly different results. Also, these embeddings may have outperformed noncontextualized embeddings because their features are already based on full sentences. As shown in Figure 2, the baseline exhibited a high level of performance, between that of noncontextualized embeddings with context and contextualized embeddings.

Comparing the results on the test (Figures 2a) and blind (Figures 2d) data sets, we notice the usual drop in performance, as both data sets differ in their distribution. The test set comes from news articles, while the blind set is based on Wikipedia. With other feature-based models submitted to the Shared Tasks, we expect this performance drop to be higher, so that our model would generalize better.

We carried out a further analysis on the test data in order to characterize weaknesses of using word embeddings for connective disambiguation. Therefore, we examined our contextualized embedding model without context (`bert-ctx-0`), as it yielded high performance despite its low complexity. For most of the rare classification mistakes made by our model, we found that there existed similar embeddings to those that were misclassified, which naturally made them hard to distinguish for our model.

4.2 Joint Disambiguation and Sense Classification

For our second experimental setting, we study the influence of jointly training connective disambiguation and sense classification (coarse and fine senses shown in Figures 2b and 2c, respectively). As our hypothesis, we assumed generalization to improve with increasing task complexity. For the commonly evaluated F1 score, we do not see a vast improvement between connective disambiguation and the joint training approach. In addition to the previous metric, we use the average precision score that better summarizes the overall ratio of precision and



(a) Coarse Sense

(b) Fine Sense

Figure 3: Confusion matrix of `bert-ctx-0` on test data for joint disambiguation and sense classification. Relative class values are reported for coarse and fine senses. The label None represents an absence of connectives.

recall for a single model. With respect to this metric, we notice higher values for both kinds of sense classification in contrast to connective disambiguation. This confirms that, although the single F1 score might not change that much, more complex tasks indeed improve model generalization and result in more stable models. Further, it appears that training our model on fine senses is somewhat less effective for the main disambiguation task, as training on coarse senses often slightly outperforms it.

Finally, we studied the predictions of our contextualized embedding model (see Figure 3) as before. Here, we compare both sense levels and notice a change of performance in Contingency and Expansion. While the coarse model works better on the second class than the first one, this turns around for the fine-sense model. Especially for the fine-sense model, we observe an overall drop of performance, which could be related to the smaller number of samples per class.

5 Discussion

In our final comparison, Table 3 shows our best-performing models for each category (standard vs. contextualized embeddings and connective disambiguation vs. joint training for sense classification). For comparison, we included test results from successful submissions to the CoNLL 2016 Shared Task (Xue et al., 2016)—in particular, results that were achieved for connective disambiguation in the first part of the Shared Task and results for explicit sense classification taken from the second part. As Table 3 shows, when using contextualized embeddings, our model outperformed the other systems with F1 scores of up to 97.32. The authors of the models (`Stepanov` and `Soochow`) unfortunately submitted results only for the first task, and thus, we cannot compare their performance on sense classification to our model's performance. The numbers for sense classification of our proposed contextualized embedding approach are slightly below those of the compared systems. But it is important to note that the other systems are prone to error propagation—errors made early throughout the pipeline negatively affect all subsequent steps. However, in the competitions, error propagation was eliminated by providing preprocessed data to the competing systems. This can be considered an unfair advantage over our system, which performed all tasks simultaneously and thus had to operate on raw data.

6 Related Work

In this section, we discuss work relevant to the area of discourse parsing, in particular, connective disambiguation and sense classification. Finally, recent work on word embeddings and multitask learning with regard to discourse parsing is outlined.

For *connective disambiguation*, Pitler and Nenkova (2009) defined a set of syntactic features extracted from constituency trees. Beside the connective's surface and category information from related tree nodes (parent, siblings), they also used

Model	Test		Blind	
	$F1_{conn}$	$F1_{sense}$	$F1_{conn}$	$F1_{sense}$
Ecnuc	93.96	90.13	91.34	77.41
OPT	94.43	90.13	91.79	77.17
Stepanov	92.43	—	88.56	—
Soochow	94.74	—	91.04	—
ctx-embd (bert-ctx-1)	97.32	—	96.98	—
ctx-embd-mtl (bert-ctx-0)	97.18	86.26	96.28	71.69
embd (wiki-ctx-1)	92.09	—	88.74	—
embd-mtl (wiki-ctx-1)	92.63	80.39	89.74	62.74

Table 3: Task-related F1 scores. Results are taken from the CoNLL 2016 Shared Task website (`http://www.cs.brandeis.edu/~clp/conll16st/results.html`) for the following parsers: OPT (Oepen et al., 2016), Ecnuc (Wang and Lan, 2016), Stepanov (Stepanov and Riccardi, 2016a), Soochow (Fan et al., 2016). The ending `-mtl` refers to the results for fine-sense classification in Table 2.

binary features that check whether categories are contained by the nodes' traces and pairwise interaction features. In addition to these features, Lin et al. (2014) propose a set of lexicosyntactic features, as they observe that a connective's immediate context and part of speech is already a strong indicator for disambiguation. The authors further extend those features by category paths from the connective to the root. Wang and Lan (2015) further extend the previous two works and add similar features for more syntactic context information of the connective. Oepen et al. (2016) combine previous feature sets with work on identifying expressions of speculation and negation (Velldal et al., 2012). Recent work of Webber et al. (2019) highlights the complexity of several kinds of ambiguity when working with discourse connectives.

The connective and its *explicit sense* have a strong correlation as shown by Pitler and Nenkova (2009), who report accuracy higher than the interannotator agreement for their connective disambiguation features on coarse-grained level senses. Lin et al. (2014) use only context features and evaluate their work on second-level senses. Wang and Lan (2015) extend previous features and develop a model for the CoNNL Shared Task. Oepen et al. (2016) use an ensemble of three types of classifiers that are mainly based on previous features (Wang and Lan, 2015). Stepanov and Riccardi (2016b) use chained information extracted from syntactical trees and chunk tags. Qin et al. (2016) use convolutional neural networks on word-level embedded sentence pairs but a linear model with additional dependency features for sense classification.

Braud and Denis (2015) have shown that word embeddings outperform sparse features for implicit sense classification. They compare word pair features with Brown clusters and low-dimensional word embeddings. Bai and Zhao (2018) use different levels of input representations, ranging from character level to contextualized word embeddings. Kishimoto et al. (2020) adapt BERT to perform implicit discourse sense classification. They show promising results by adding tasks, such as connective prediction, for pretraining.

Multitask learning is also successfully applied to implicit sense classification (Liu et al., 2016). The authors combine four different tasks related to discourse parsing, but in contrast to our work, they rely on previously extracted argument spans. Qin et al. (2017) propose a model that, in addition to their main task (implicit sense classification), also learns to predict a possible connective that could be inserted. Lan et al. (2017) introduce various models that perform multitask learning, and their focus also lies on implicit sense classification.

7 Conclusions

In this work, we studied the value of discourse information in different kinds of word embeddings. We first presented a novel feature-free approach to connective disambiguation that achieves state-of-the-art results on this task. Then, this approach was extended by explicit sense classification to study the influence of jointly training both tasks. While our second approach does not directly outperform previous approaches on explicit sense classification, our model can be directly applied to raw input without being subject to error propagation, which is an advantage of our approach.

As our work indicates that combining multiple subtasks avoid error propagation issues, a future direction could be to investigate what other kinds of subtasks could be combined in order to benefit from this. Also, word embeddings have shown to be very flexible, and they are useful even for out-of-domain data. It is worth investigating whether they are suitable for language transfer. This is particularly interesting because data sets of a similar quality to PDTB do not exist for many languages other than English.

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.

Jonathan Baxter. 2000. A model of inductive bias learning. *Journal of artificial intelligence research*, 12:149–198.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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.

Richard A. Caruana. 1993. Multitask learning: A knowledge-based source of inductive bias. In *Machine Learning Proceedings 1993*, pages 41 – 48. Morgan Kaufmann, San Francisco (CA).

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12(null):2493–2537.

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.

Ziwei Fan, Zhenghua Li, and Min Zhang. 2016. Finding arguments as sequence labeling in discourse parsing. In *Proceedings of the CoNLL-16 shared task*, pages 150–157. Association for Computational Linguistics.

Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. 2011. Shallow discourse parsing with conditional random fields. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1071–1079, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Yudai Kishimoto, Yugo Murawaki, and Sadao Kurohashi. 2020. Adapting BERT to implicit discourse relation classification with a focus on discourse connectives. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1152–1158, Marseille, France. European Language Resources Association.

Man Lan, Jianxiang Wang, Yuanbin Wu, Zheng-Yu Niu, and Haifeng Wang. 2017. Multi-task attention-based 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.

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

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Yang Liu, Sujian Li, Xiaodong Zhang, and Zhifang Sui. 2016. Implicit discourse relation classification via multi-task neural networks. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2750–2756. AAAI Press.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Stephan Oepen, Jonathon Read, Tatjana Scheffler, Uladzimir Sidarenka, Manfred Stede, Erik Velldal, and Lilja Øvrelid. 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, Berlin, Germany. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. 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 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Emily Pitler and Ani Nenkova. 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. 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 *In Proceedings of LREC*.

Rashmi Prasad, Bonnie Webber, and Alan Lee. 2018. Discourse annotation in the PDTB: The next generation. In *Proceedings 14th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, pages 87–97, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

L. Prechelt. 1998. Automatic early stopping using cross validation: quantifying the criteria. *Neural networks : the official journal of the International Neural Network Society*, 11 4:761–767.

Lianhui Qin, Zhisong Zhang, and Hai Zhao. 2016. Shallow discourse parsing using convolutional neural network. In *Proceedings of the CoNLL-16 shared task*, pages 70–77, Berlin, Germany. Association for Computational Linguistics.

Lianhui Qin, Zhisong Zhang, Hai Zhao, Zhiting Hu, and Eric Xing. 2017. Adversarial connective-exploiting 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.

Karin Sim Smith. 2017. On integrating discourse in machine translation. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 110–121, Copenhagen, Denmark. Association for Computational Linguistics.

Evgeny Stepanov and Giuseppe Riccardi. 2016a. Unitn end-to-end discourse parser for conll 2016 shared task. In *Proceedings of the CoNLL-16 shared task*, pages 85–91. Association for Computational Linguistics.

Evgeny Stepanov and Giuseppe Riccardi. 2016b. UniTN end-to-end discourse parser for CoNLL 2016 shared task. In *Proceedings of the CoNLL-16 shared task*, pages 85–91, Berlin, Germany. Association for Computational Linguistics.

Erik Velldal, Lilja Øvrelid, Jonathon Read, and Stephan Oepen. 2012. Speculation and negation: Rules, rankers, and the role of syntax. *Computational Linguistics*, 38(2):369–410.

Jianxiang Wang and Man Lan. 2015. A refined end-to-end discourse parser. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*, pages 17–24, Beijing, China. Association for Computational Linguistics.

Jianxiang Wang and Man Lan. 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.

Longyue Wang, Chris Hokamp, Tsuyoshi Okita, Xiaojun Zhang, and Qun Liu. 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.

Bonnie Webber, Rashmi Prasad, and Alan Lee. 2019. Ambiguity in explicit discourse connectives. In *Proceedings of the 13th International Conference on Computational Semantics - Long Papers*, pages 134–141, Gothenburg, Sweden. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Yuxiang Wu and Baotian Hu. 2018. Learning to extract coherent summary via deep reinforcement learning. *CoRR*, abs/1804.07036.

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. Association for Computational Linguistics.

Nianwen Xue, Hwee Tou Ng, Sameer Pradhan, Attapol Rutherford, Bonnie Webber, Chuan Wang, and Hongmin Wang. 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.

Yang Zhong, Chao Jiang, Wei Xu, and Junyi Jessy Li. 2020. Discourse level factors for sentence deletion in text simplification. In *AAAI*.