Towards Identifying Alternative-Lexicalization Signals of Discourse Relations

René Knaebel and **Manfred Stede**
Applied Computational Linguistics
Department of Linguistics
University of Potsdam
Germany
{rene.knaebel,manfred.stede}@uni-potsdam.de

Abstract

The task of shallow discourse parsing in the Penn Discourse Treebank (PDTB) framework has traditionally been restricted to identifying those relations that are signaled by a discourse connective ("explicit") and those that have no signal at all ("implicit"). The third type, the more flexible group of "AltLex" realizations has been neglected because of its small amount of occurrences in the PDTB2 corpus. Their number has grown significantly in the recent PDTB3, and in this paper, we present the first approaches for recognizing these "alternative lexicalizations". We compare the performance of a pattern-based approach and a sequence labeling model, add an experiment on the pre-classification of candidate sentences, and provide an initial qualitative analysis of the error cases made by both models.

1 Introduction

The view that discourse relations serve to model central aspects of the coherence of a text is widely accepted, and several approaches with different theoretical commitments have been developed (e.g. Mann and Thompson, 1988; Prasad et al., 2008a; Lascarides and Asher, 2007; Sanders et al., 1992). Our work is situated in the framework of Shallow Discourse Parsing and thus grounded in the Penn Discourse Treebank (PDTB) corpus (Prasad et al., 2008a). Here, some distinctions are commonly being made regarding the surface realization of discourse relations; most importantly:

1. A relation can be signalled by a *connective*, i.e., a lexical item from a closed class (conjunctions, certain adverbials).
2. A relation can be signalled by a different lexical form, which the PDTB calls *Alternative lexicalization* or *AltLex* for short.
3. A relation can also be stated without any lexical signal; in this case it is called *implicit*.

In the PDTB corpus, (1) and (3) are by far the most frequent cases, and accordingly, they have received much attention in shallow discourse parsing. As for (2), Lin et al. (2014) had developed a first, relatively simple approach; to our knowledge there have not been any follow-up proposals (including all the parsers presented in the CoNLL shared tasks in 2015 and 2016 (Xue et al., 2015, 2016) and in the recent DISRPT tasks (Zeldes et al., 2019, 2021)).

With the introduction of the PDTB corpus version 3.0 (Prasad et al., 2018), amongst some other changes, the number of annotated AltLex instances has grown from 624 (in version 2.0) to 1632. With the corpus now being considerably richer in AltLex signals, we believe that their role in shallow discourse parsing now needs to be strengthened. Besides, also for advancing the theoretical description of the AltLex category and its role in coherence marking, it is important to perform empirical studies.

Essentially, alternatively–lexicalized discourse relations are signalled by an open set of phrases that verbalizes the connection between two discourse arguments. In the PDTB, they are being annotated when no connective is present, and a "connective insertion" test yields an impression of redundancy; in this case annotators are asked to mark the text span that already signals the relation. Previous work (Prasad et al., 2010; Danlos, 2018; Rysová and Rysová, 2018) studied the general form of AltLex expressions and tried to find and formalize patterns that can complement the well-established idea of a fixed list of discourse connectives (e.g. Das et al., 2018).

Example (1) (Danlos, 2018) illustrates the interchangeability of explicit connectives and AltLex signals withing the same context, as we could simply substitute the connective *Therefore* by alternative more complex lexicalizations such as *This caused* and *Because of this* while still preserving the meaning of the relation:

Proceedings of the 29th International Conference on Computational Linguistics, pages 837–850
October 12–17, 2022.

(1) 1. Fred didn't stop joking. **Therefore**, his friends enjoyed hilarity throughout the evening.
2. . . . **This caused**, his friends enjoyed hilarity throughout the evening.
3. . . . **Because of this**, his friends enjoyed hilarity throughout the evening.

Contrary to this, in Example (2) (Prasad et al., 2010) the AltLex signal is not easily substitutable by a simpler explicit connective as, with a replacement such as *because*, this sentence would lose information about the reason's importance:

(2) But a strong level of investor withdrawals is much more unlikely this time around, fund managers said. **A major reason** is that investors already have sharply scaled back their purchases of stock funds since Black Monday.

In this paper, we aim to overcome the negligence of AltLex relations in shallow discourse parsing by proposing two different technical approaches. Both tackle the problem without relying on any external lexical resources. Specifically, our contributions are: (i) We present the first approach to automatically classifying AltLex instances in the PDTB3 corpus; (ii) we compare a simple pattern-based approach to a neural model; (iii) we experiment with pre-classification of AltLex-relevant sentences (for dealing with class imbalance in the corpus); (iv) we provide initial observations on types of errors made by the models.

Section 2 discusses related work, and Section 3 briefly describes relevant aspects of the PDTB corpus. Section 4 introduces our various methods. We present results in Section 5, discuss them in Section 6, and finally conclude.

2 Related Work

We highlight two different directions that are relevant for our work. The first part shows work related to alternative lexicalized phrases. The second one looks into recent applications for connective identification, whose methods are similar to one of our approaches.

Alternative Lexicalized Phrases (AltLex). After the introduction of alternative lexicalized phrases, among others, in the PDTB (Prasad et al., 2008a), a subsequent work by Prasad et al. (2010) presented more details and analysed regularities by defining groups of phrases. Based on their discoveries, a first attempt was made to extract AltLex relations while analysing implicit relations (Lin et al., 2014). Their approach is evaluated in combination with all non-explicit relations. Thus, the specific AltLex results of their approach are unfortunately not available for comparison. Due to the revision of the PDTB, the definition of discourse signals is made more flexible, and Lin et al.'s approach is not applicable to the current problem anymore (Prasad et al., 2018).

Attempts of building a Czech Discourse Treebank have also shown, how challenging the annotation process is. In contrast to the PDTB, the Czech Discourse Treebank distinguishes discourse signals into three groups, namely *primary connectives*, *secondary connectives*, and *free connective phrases*. Low inter-annotator agreement was particularly observed for the annotation of free connecting phrases (comparable to alternative lexicalizations) due to the complexity of the task (Rysová, 2012). Based on this tree bank, a template approach is proposed to manually build a lexicon for secondary connectives, analogously to that for primary ones (Danlos, 2018; Rysová and Rysová, 2018). One of Danlos's single lexicon entry, for example, to recognize the phrases `for this/a given reason`, would describe the lexical head `N` of this phrase `reason` in different possible environments (so-called schemes) with a rule like `for [Ana-Det (Adj)/Ana-Adj] N`. Furthermore, Danlos (2018) concedes that, because "free connective phrases are compositional and include at least two content words", this lexicon-based approach is not applicable.

Dunietz et al. (2017) adapt the PDTB annotation scheme and present another corpus that entirely focuses on causal relations. They do not distinguish between explicit connectives and alternative lexicalizations as done in the PDTB. Comparing their annotated signals with the second version of PDTB, they discover an overlap of 8.9 % with the PDTB connective signals. Further, they introduce a feature-based system for tagging causal relations between individual events. In contrast to them, we avoid linguistic features by using representations from pretrained language models.

In contrast to PDTB conform schemes, RST (Mann and Thompson, 1988) contains only information about discourse segments, their relations and nuclearity. The work of Das and Taboada

(2018) complements the absence of this information by annotating a subpart of the RST-DT corpus with all kinds of signals point toward a discourse relation. Their first finding reveals that discourse relations may be signalled by other discourse elements than lexical phrases, *inter alia*, syntactic structure (e.g. relative clauses, reported speech), semantics (e.g. synonymy, repetition, lexical chain), text genre (e.g. inverted pyramid scheme, newspaper layout). Another interesting finding consists in the presence of multiple signals that point to the same relation, e.g. semantic+syntactic, reference+syntactic, and others. Further, recent work of Zeldes and Liu (2020) proposed an interesting approach for inferring discourse signals from the given relation senses. However, this approach does not take alternative lexicalizations into account yet.

Explicit Connective Disambiguation. Some of the approaches that were proposed to this task are relevant for AltLex identification as well. Pitler and Nenkova (2009) presented a simple feature-based model for the disambiguation (discourse versus sentential usage) of connective candidates extracted by matching entries of a connective lexicon. With their best feature combination, they achieve an F1 score of 94.19 on the test set. Recently, Knaebel and Stede (2020) adapt the original idea and replace hand-crafted features by various types of word embeddings. Their state-of-the-art model uses contextualized word embeddings and achieves 97.45 F1 score. In our work, we extend their approach to alternative lexicalized phrases, tackling the problem without external candidate lexicon.

Furthermore, another line of research started with the focus to avoid lexicon-based solutions. Recently, several promising sequence labeling approaches (e.g. Yu et al., 2019; Muller et al., 2019; Bakshi and Sharma, 2021; Kamaladdini Ezzabady et al., 2021) have been proposed using standard and contextualized word embeddings. Among these, Yu et al. (2019) achieves best scores (92.02 F1 score) in extracting connectives without lexicon. They develop a model that combines linguistic information (e.g. part-of-speech tags, dependency relation, sentence length, *inter alia*) with recent advances in contextualized word representations. In contrast, the work of Bakshi and Sharma (2021) achieve slightly worse results (91.15 F1 score) but with a completely feature-free approach.

3 Penn Discourse Treebank

The recent version (v3) of the Penn Discourse Treebank (PDTB, Prasad et al. 2018) is the largest available resource of lexically grounded discourse relations which include both explicitly signalled relations and implicit relations. It describes discourse relations to consist of exactly two arguments with an optional marker to signal the relation. In addition, one or more *senses* are attached to each relation to describe its meaning. For example, two senses often correlate are *Temporal.Synchronous* and *Comparison.Contrast* for the explicit connective *while*.

In its previous version, Prasad et al. (2008b) start the annotation process by the identification of connectives (defined by a fixed set of candidates). Then, adjacent sentences without explicit relation are examined according to whether there holds an implicit relation, and, in addition, a connective is searched that fits in between the relations' arguments. If the insertion of any connective leads to redundancy, the lexical signal already part of the relation is used instead—the AltLex. In their studies on alternative lexicalizations, Prasad et al. (2010) use the two properties of syntactical and lexical flexibility to sort these signals into three groups. Hereby they demonstrated, that most of the AltLexes belong to the syntactically and lexically free group with 76.6%.

As a consequence of the recent update, the definition of discourse signals has undergone some changes. The set of explicit connectives is expanded (which indirectly changes the set of AltLexs too) and the position of the connective in relation to its arguments is more relaxed compared to the previous version. In addition, a few changes have been made in the process of identifying AltLex relations in general. As a result of the introduction of intra-sentential AltLex relations (about 900), arbitrary expressions are allowed, also including adjectives and adjective modifiers. Also, annotators are allowed to mentally add anaphoric references, (e.g. next [to this], further [to that]), which leads to potential overlap with explicit connectives. Signals are not syntactically bound to the second argument of the relation anymore, but possibly combine parts of both arguments. A new sub-class was introduced for lexico-syntactic constructions, the so-called AltLexC. In total, almost 1000 AltLex relations, including their signals, were added during the revision. In our work, we do

Property	PDTB2	PDTB3
count	624	1632
signal length	3.26 (2.08)	2.62 (2.45)
sentence length	22.65 (10.35)	27.68 (10.94)
signal position	1.42 (3.22)	9.37(10.26)

Table 1: Differences of AltLex relations between both versions of the Penn Discourse Treebank. Properties length and position show average values with standard deviation enclosed in parentheses.

not further distinguish both types of alternative lexicalized relations and henceforth refer to these simply as AltLex relations. Discourse signals appear in both continuous (e.g. *since then, after that*) and discontinuous forms (e.g. *the aim . . . is*, *the more . . . the more*, ...). As the complexity highly increases if POS patterns would also cover gaps within, we restrict our work to continuous signals only, and thus eliminated 62 instances (3.79 %) from the full set. Explicit discourse connectives are ignored for all our experiments, even if they should occur in the same sentence of an AltLex signal.

The PDTB consists of 2,160 documents with a total number of 50,945 sentences. We briefly summarize the differences of both versions of the PDTB in Table 1 to illustrate the motivation of introducing our new approaches. As already mentioned, the number of available AltLex relations is almost tripled from 624 to 1632 instances. The average length of the signals decreases slightly from 3.26 tokens to 2.62 token, and the average length of sentences containing the signals increases by a few tokens on average (22.65 up to 27.68). A particular challenging aspect of the new PDTB version is the more flexible positioning of signals, which renders previous simple identification approaches as no longer feasible.

4 Method

Input to all models are prepared context sensitive word embeddings. For extracting token-wise context sensitive embeddings, we follow the suggestion of Devlin et al. (2019). Given a sentence, tokens are processed by the WordPiece tokenizer (Wu et al., 2016) which possibly leads to a higher number of subtokens. These are processed for generating corresponding hidden states on subtoken-level.

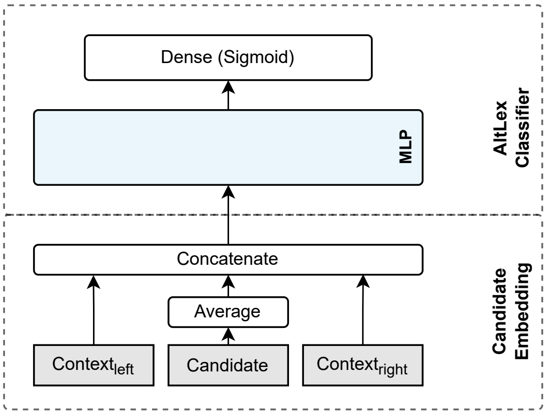


Figure 1: Candidate-based classification approach. All token embeddings directly associated with AltLex candidate are averaged and concatenated with surrounding context tokens. Simple MLP module on top with final classification layer to predict discourse signal.

We choose RoBERTa (Liu et al., 2019) as it performs best on connective disambiguation compared with other BERT variations (Knaebel and Stede, 2020). We average multiple subtoken outputs into a single output that corresponds to the full token. Following the suggestion of Devlin et al. (2019), the last four hidden layers are concatenated and thus form the final token embeddings that serve as input for the subsequent models.

4.1 Pattern-based Candidate Extraction

Traditional approaches for connective disambiguation integrate a connective lexicon that is used to extract possible candidates, before using a system to disambiguate discourse readings from their sentential counterparts (Pitler and Nenkova, 2009; Lin et al., 2014). Inspired by this approach, we devised a different pattern-based extraction procedure for finding possible AltLex candidates. Specifically, we generate possible AltLex candidates by pattern-matching via a list of extracted part-of-speech (POS) sequences. From the AltLex relations available in PDTB3, we extract 408 unique POS patterns. They range from very frequent single tags such as **VBG** (n=468) and **RB** (n=113) to longer and less frequent sequences such as **VBD DT NN IN** (n=9). This sequence, for example, is extracted from the signals *attributed the increase/improvement to*, but it also matches other phrases such as *visited a lot of* and *signed a contract with*. The number of extracted candidates per pattern is very high in comparison with the pattern occurrence itself. For the simple single tag rule **VBG**, for example, we find 1869 instances in one of our randomly sampled test split.

We extend this basic pattern approach by introducing a small context window of additional tokens to the left and right of the candidate. Our hypothesis is that *context sensitivity* is not only beneficial for the final classification of the embeddings, but already useful during the pattern extraction. Information such as start and end of sentence, but also punctuation is useful. As the number of patterns increases tremendously with each additional surrounding word, and the generalization at the same time decreases by patterns that are too specific, we limit our experiments to only one tag on the left and on the right of the original pattern. By doing so, the number of extracted patterns increases to 800 in total. For example, instead of one most frequent single-tag pattern **VBG**, we now have more specific patterns that occur less often **, VBG DT** (n=162), **, VBG NNS** (n=39), **, VBG PRP** (n=35), **, VBG NN** (n=31), among others. In comparison to the simple approach, the context-sensitive approach reduces the number of extracted candidates by a large margin, in particular for shorter tag patterns. For example, the number of occurrences for **VBG** is decreased by about 70 % to 516 instances. A list of the 30 most frequent tag sequences for both approaches is provided in the appendix Table 4, and Table 5 provides numbers on the extracted candidates per approach.

We use the collected tag lists, iterate over all sentences, and extract any phrase that matches one of the POS patterns as possible AltLex candidate. After generating these candidates, we follow the approach of Knaebel and Stede (2020) for connective disambiguation with contextualized embeddings, shown in Figure 1. In their experiments, they outperform previous approaches with a simple multilayered-perceptron architecture on top of contextualized embeddings. We refer to this architecture as *MLP* module, which consists of two fully connected layers with a dropout layer following each.

The first ("simple") approach is henceforth referred to as **exact** approach, while we call the second one **context-sensitive**. For the experiments, we also specify the **context size**, that is the number of tokens surrounding the candidate on each side; e.g., the value 0 refers to no context at all, while the value 2 indicates a context of two tokens to the left and right, which sums up to five embeddings (four context embeddings plus one embedding for the averaged candidate tokens).

4.2 Sentence Labeling

The limited variability of observed patterns in the dataset is the major disadvantage of the pattern-based approach. We aim to overcome this problem by introducing a **sequence labeling** approach (see Figure 2) based on contextualized embeddings for recognizing alternative lexicalizations. In the PDTB3, AltLex signals always occur within a single sentence, and thus our approach is designed for sentence-level processing.

The sentence processing part consists of two bidirectional LSTM (Hochreiter and Schmidhuber, 1997) layers. After each layer we add a dropout layer for better generalization. We will refer to this as *BiLSTM* module. The hidden states are further individually processed by an MLP module. Finally, we use a conditional random field (Lafferty et al., 2001) for the output prediction (compare Figure 2 output (I) Sentence Labeling). As output, we use a binary label that represents the AltLex class membership.

Deciding whether a phrase should be identified as AltLex is often dependent on its context. As argument spans of AltLex relations often include previous sentences, we hypothesize that additional processing of the preceding sentence has a positive effect on the prediction quality of our model. For this reason, we propose an additional sentence processing step (see Figure 2 **Context Processing** on the left side) in which we use the final hidden states of the BiLSTM module as the initial states for the BiLSTM modules in the sentence processing part.

We refer to the architecture without previous sentence context as **single**, and we use **context** to point to the option with previous sentence processing.

4.3 Sentence Classification

The extraction of alternative lexicalized discourse signals is especially hard with respect to the small amount of signal occurrence. In addition to the heavy imbalance of the labels on token-level, only a minority of all sentences contains an alternative lexicalized signal. Therefore, we hypothesize that, following analogously work related to explicit connectives (Patterson and Kehler, 2013), an additional step of classifying a potential sentence candidate as containing an AltLex relation or not might be

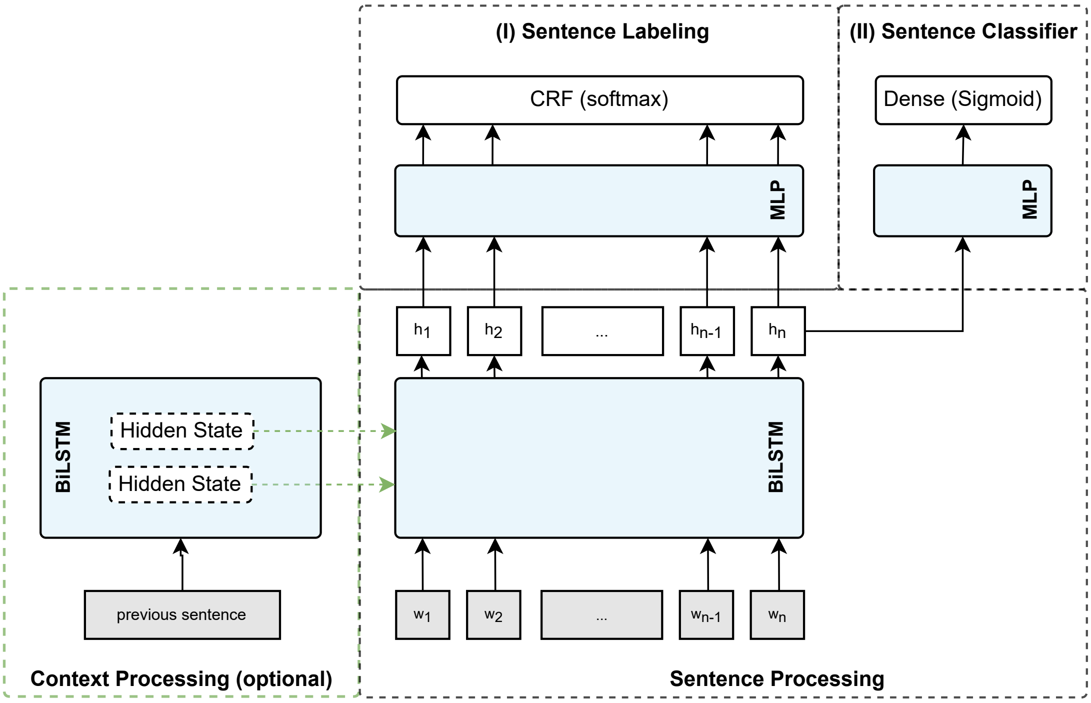


Figure 2: Joint diagram of (I) sequence labeling and (II) potential sentence classification approaches. A given sentence is processed by BiLSTM module. Either all hidden states are processed by MLP module individually and forwarded to CRF layer to predict AltLex tokens in sentence, or only last hidden state is processed by MLP module to predict the presence of AltLex in whole sentence. The optional context processing part contains a similar BiLSTM module but processes the previous sentence. Then, hidden state of the LSTMs are used for initialization.

beneficial to reduce the overall complexity of the sentence labeling problem. We first train a **sentence classification** model on the full dataset, including the majority of negative examples. Thereafter, we train our sentence labeling architectures (as described above) on the positive instances only.

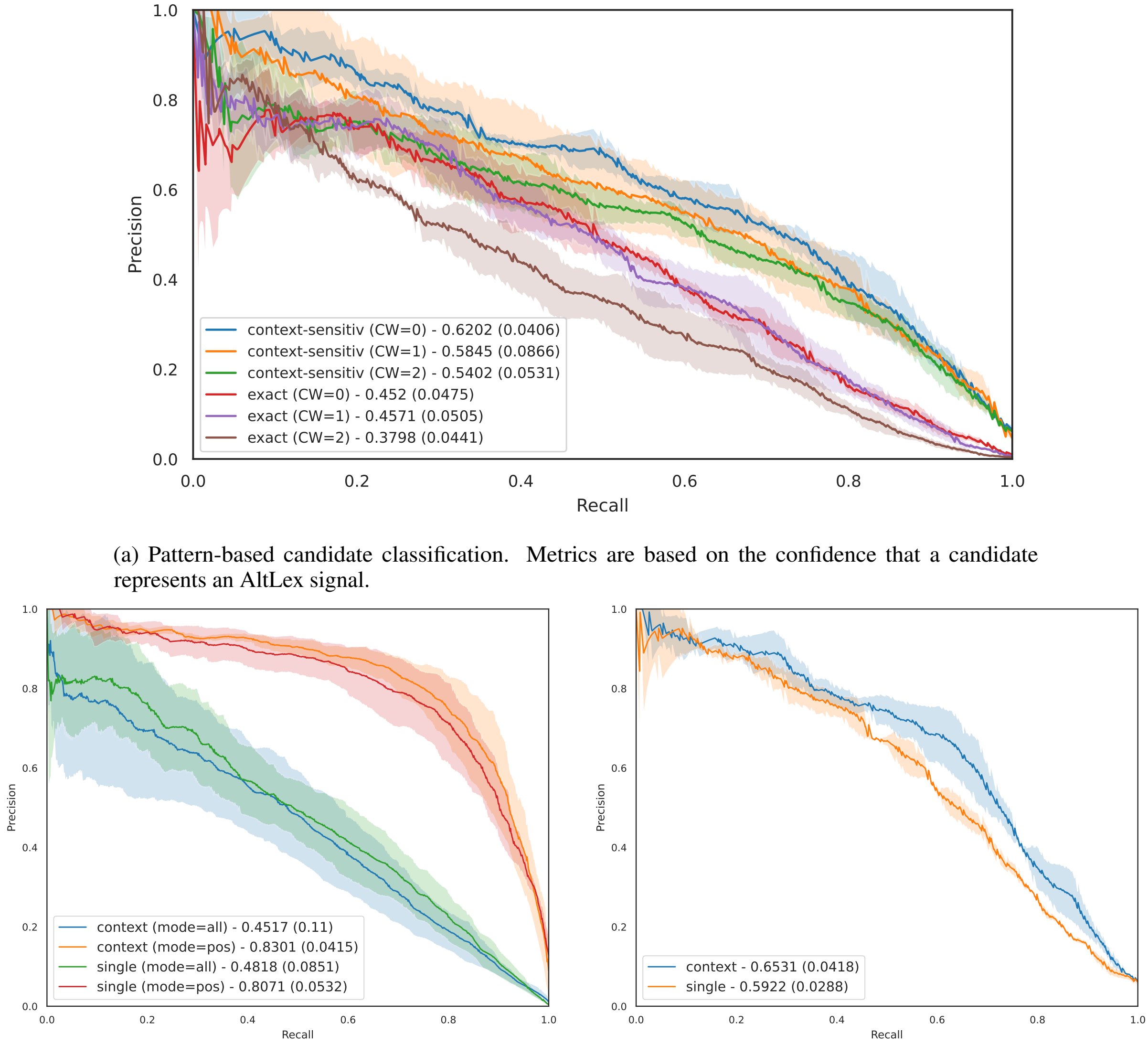
5 Experiments

In our experiments we study various statistical models on version 3 of the Penn Discourse Treebank (Prasad et al., 2018). For all experiments' runs, we randomly split the full dataset into three parts (train, validation, and test), as suggested by Shi and Demberg (2017). We set 10 % of the full dataset aside for testing, then the remaining data is split into 90 % and 10 % for training and validation parts, respectively. The reported final results are averaged over three different runs each. For the precision–recall curves illustrated in Figure 3 we interpolate the individual curve per run and compute the mean curve surrounded by one standard deviation. We use the average precision scores (AP) of the mean curves for comparison.

5.1 Pattern-based Disambiguation

We compare two variants of our pattern-based extraction, **exact** patterns and **context-sensitive** patterns, taking into account one token to the left and right of the pattern. We generate patterns for both variants only once on the whole corpus which we think is most similar to the experiments on connective classification, where a list of possible connective candidates is compiled previously before splitting data. Further, we study the influence of the number of surrounding context embeddings for the model (context width). Through all experiments, we use an up-sampling rate of 5 for positive samples and 0.1 for negative samples. The hidden size of the first layer in the MLP module is 256 and the second layer 64, respectively. We train for at most 20 epochs with a batch size of 64. In addition, we stop earlier if validation loss does not improve over 7 epochs.

Evaluation metrics are calculated with regard to the extracted signal candidates. Please note, that in contrast to the later evaluation (Section 5.3), multiple possibly overlapping candidates might be extracted. Analysing the precision–recall curves in Figure 3a, an increased context width for both



(a) Pattern-based candidate classification. Metrics are based on the confidence that a candidate represents an AltLex signal.

(b) Sentence Labeling approach. Metrics are computed token-wise based on the potentials of the CRF layer before optimal sequence is computed.

(c) Sentence Classification. Metrics computed with respect to confidence of the model that a sentence contains AltLex signal.

Figure 3: Precision–Recall curves for all trained models separated by tasks. Curves of individual run are averaged surrounded by one standard deviation. Scores represent average precision scores with standard deviation in parenthesis.

pattern-based models seems not beneficial, as the context is already encoded in the individual token embedding. Using context-sensitive patterns for extraction, on the other hand, has a positive influence, as originally assumed. On the one side, the overall number of unique extracted patters roughly doubles due to the higher specification. On the other, the total number of extracted candidates reduces by about 95% on a random test split. Finally, the best performing model in this section (context-sensitive pattern, zero embedding context width) achieves a token-level averaged score of 0.62 precision.

5.2 Sentence Labeling

A sequence-level labeling model is trained to predict the presence of an discourse signal. The model assumes as input contextualized word embeddings for a maximum sentence size of 60 tokens. For single-step prediction, we use a down-sampling rate of 0.5, for two-step prediction we remove negative samples entirely. All modules' layers use 128 units as hidden size. We give all models the chance to train for 50 epochs with a batch size of 32; however, the training stops earlier in all cases, when the validation loss stops improving over 7 epochs.

For the sentence labeling approach, we use token-wise potentials of the CRF output instead of the decoded sequence labels. This allows us to calculate precision and recall on token-level which is visualized in Figure 3b. We observe that additional context in form of the previous sentence is not beneficial on this level for sentence labeling. With a mean score of 0.45 AP, the performance of the context-sensitive labeling model is slightly below the score of the single sentence labeling model with 0.48 AP (note the low confidence caused by the high standard deviation). The performance increases dramatically, as expected, when labeling is restricted to positive sentences only.

For potential sentence classification, the previous sentence's context increases the average performance slightly (from 0.59 to 0.65 average precision) as indicated by Figure 3c. This is in accordance with the previous observations for the sequence labeling experiment.

5.3 Results

For the final evaluation, we introduce the metrics *overlap*, *partial-rate*, and *full-rate*. The first score indicates the overlap of true signal positions with predicted signal occurrences. The partial-rate is one if there is at least one token of the signal correctly classified. The full-rate is satisfied if and if only the full range of the signal is correctly recognized. The sequential predictions are taken as computed by the final layer, for the candidate-based predictions, we simply choose all as signal classified candidates and set corresponding associated tokens to being a signal. Note that a single token might be classified multiple times. A token is set to be a signal if a single instance prediction exists.

Results of the experiments are presented in Table 2a for the candidate-based experiments and in Table 2b for the labeling and sentence classification experiments. Scores for precision, recall, and F1 with respect to predicting the AltLex class are presented for a 0.5 threshold. Thus, they merely provide a limited view compared to the precision–recall curves.

The best overall performing model is the candidate-based model with context-sensitive pattern and zero embedding context width. It achieves a 74 % phrase overlap, with scores 0.83 and 0.63 for partial-rate and full-rate, respectively. Interestingly, the candidate-based approach outperforms the sequence-labeling approach. Although the F1 scores are higher for this approach, the final evaluation shows that lower performance regarding overlap (0.63), partial-rate (0.67), and full-rate (0.60). The performance increases dramatically, as expected, when labeling is restricted to positive sentences only. The combination of the simple sentence classification (0.64 F1 score) and the simple labeling approach on positive samples (0.84 F1 score) leads to similar results as a model trained on the full data set.

6 Discussion

Both pattern-based candidate approaches (exact and context-sensitive) achieve better final results compared to the labeling approaches which reflects a similar observation as for studies on explicit connective identification (Knaebel and Stede, 2020). However, we have to keep in mind that patterns are extracted on the whole corpus in advance, which makes the lexicon approach for explicit connective identification more similar and, thus, better comparable. The comparison to our sequence labeling approach is somewhat unfair, as we here strictly split the corpus into three parts right at the beginning and these models never have access to all signal variants in the whole corpus.

Compared to the recognition performance of explicit connectives with about 96%, AltLex relations are predicted far less accurately with at most 63% exact match for the pattern-based approach. Also, we expect a drop in performance when limiting the extraction to the training corpus only.

After examining the errors made by the pattern-based mode, we conclude that the length of the errors (unrecognized signals) compared to the correctly recognized patterns is almost the same (2.5 and 2.7 tokens on average each). Thus the length of signals is not a crucial factor for this type of model. Quite a few poorly recognized examples are related to adverbial phrases (e.g., *eventually, further, so far, too*). This type of error seems problematic for both model variants. A possible reason is the high imbalance of adverbial signal instances (about 100), compared to extracted candidates ranging from 1000 to 4000 instances. Regarding verb gerund forms, the pattern-based approach recognizes most of them. This is interesting as it represents the largest group in PDTB with most variations. On the other side, we also recognize cases where signals are marked as false positives, such as *resulting* and *allowing*. Here, more more in-depth elaboration is necessary, to check whether the model is truly wrong or just

model-type	context-size	precision	recall	F1	overlap	partial-rate	full-rate
exact	0	0.29	0.58	0.38	0.68	0.82	0.55
exact	1	0.30	0.53	0.38	0.57	0.72	0.43
exact	2	0.30	0.47	0.36	0.56	0.73	0.42
context-sensitive	0	0.35	**0.72**	**0.47**	**0.74**	**0.83**	**0.63**
context-sensitive	1	**0.36**	0.64	0.46	0.72	0.82	0.60
context-sensitive	2	0.35	0.60	0.44	0.67	0.78	0.56

(a) Results of candidate-based extraction approach.

model-type	input	mode	precision	recall	F1	overlap	partial-rate	full-rate
labeling	single	all	0.53	0.55	**0.53**	**0.63**	**0.67**	**0.60**
labeling	context	all	**0.59**	0.48	**0.53**	0.58	0.60	0.57
labeling	single	positives	0.84	0.85	0.84	0.87	0.90	0.82
labeling	context	positives	0.80	0.79	0.79	0.80	0.85	0.72
sentence	single	all	0.74	0.57	0.64			
sentence	context	all	0.68	0.49	0.56			

(b) Results of the sequence-labeling and sentence classification approaches are computed on token-level.

Table 2: Evaluation results: Scores on the left (precision, recall, F1) with respect to AltLex class. These scores of both tables cannot be compared directly, as prediction level differs (candidates vs. tokens). Final signal extraction on the right evaluates predictions by degree of overlap, and agreement on partial and full prediction.

found new signals.

For the sequential model, it is noticeable that prepositions are missing in the predicted signal e.g. *only to* and *opposed to*. Also, verbs in combination with anaphoric pronouns e.g. *that would leave*, *this creates*, are often recognized as discourse signal although there are not annotated as such.

The sequential models with additional context information in form of the previous sentence result in unexpectedly low performances. Intuitively, because AltLexes often connect parts of two consecutive sentences, we would expect a model's performance to increase if it gets access to more information. We assume poor performances are caused by context representation and therefore we wonder whether models especially designed to serve sentence representations would lead to better results in this experiments.

7 Conclusion

Our work is a successful first attempt to fully automatically (without hand-crafted rules) extract alternative lexicalized discourse relation signals. For this task, we propose two technically different solutions: First, a pattern-based approach working analogously to lexicon-based connective disambiguation approaches, and second, a sequence labeling approach similar to recent connective labeling approaches without external lexicon. We evaluated these models directly on their corresponding training task and, further, provide more details on the actual recognition task.

We wonder, how these two model architectures perform on a different corpus domain such as biomedical data (Prasad et al., 2011) and whether the pattern-based limitation of the first approach is noticeable. Having the new version of the PDTB with about three times as much data as before, it still seems the performance of the sequence-labeling approach is strongly limited by the amount of available data. For future work, it would be interesting to extract different patterns for generating candidates for the first approach. Universal part-of-speech tags would have the advantage of being a little more flexible (less specific word classes) while at the same time, it could be possible to use similar techniques for other languages when the embeddings model is changed to a different language or to a multi-lingual model. The sequential approach has the advantage to be able to find new patterns without observing them in the training data directly. With more raw data from possibly different domains, it would be interesting to apply this technique and examine new/other variants of alternative lexicalizations that do not occur in such form in the original corpus.

8 Acknowledgments

This research has been supported by the German Research Foundation (DFG) with grant number 455911521, project “LARGA” in SPP “RATIO”. We would like to thank the anonymous reviewers for their valuable feedback.

References

Sahil Bakshi and Dipti Sharma. 2021. A transformer based approach towards identification of discourse unit segments and connectives. In *Proceedings of the 2nd Shared Task on Discourse Relation Parsing and Treebanking (DISRPT 2021)*, pages 13–21, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Laurence Danlos. 2018. Discourse and lexicons: Lexemes, MWEs, grammatical constructions and compositional word combinations to signal discourse relations. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 30–40, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Debopam Das, Tatjana Scheffler, Peter Bourgonje, and Manfred Stede. 2018. Constructing a lexicon of English discourse connectives. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 360–365, Melbourne, Australia. Association for Computational Linguistics.

Debopam Das and Maite Taboada. 2018. RST signalling corpus: a corpus of signals of coherence relations. *Language Resources and Evaluation*, 52:149–184.

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.

Jesse Dunietz, Lori Levin, and Jaime Carbonell. 2017. Automatically tagging constructions of causation and their slot-fillers. *Transactions of the Association for Computational Linguistics*, 5:117–133.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Morteza Kamaladdini Ezzabady, Philippe Muller, and Chloé Braud. 2021. Multi-lingual discourse segmentation and connective identification: MELODI at disrpt2021. In *Proceedings of the 2nd Shared Task on Discourse Relation Parsing and Treebanking (DISRPT 2021)*, pages 22–32, Punta Cana, Dominican Republic. Association for Computational Linguistics.

René Knaebel and Manfred Stede. 2020. Contextualized embeddings for connective disambiguation in shallow discourse parsing. In *Proceedings of the First Workshop on Computational Approaches to Discourse*, pages 65–75, Online. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.

Alex Lascarides and Nicholas Asher. 2007. Segmented discourse representation theory: Dynamic semantics with discourse structure. In H. Bunt and R. Muskens, editors, *Computing Meaning: Volume 3*, pages 87–124. Kluwer Academic Publishers.

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2014. A pdtb-styled end-to-end discourse parser. *Nat. Lang. Eng.*, 20(2):151–184.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text & Talk*, 8:243 – 281.

Philippe Muller, Chloé Braud, and Mathieu Morey. 2019. ToNy: Contextual embeddings for accurate multilingual discourse segmentation of full documents. In *Proceedings of the Workshop on Discourse Relation Parsing and Treebanking 2019*, pages 115–124, Minneapolis, MN. 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 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. 2008a. The penn discourse treebank 2.0. In *In Proceedings of LREC*.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008b. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference* *on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Rashmi Prasad, Aravind Joshi, and Bonnie Webber. 2010. Realization of discourse relations by other means: Alternative lexicalizations. In *Coling 2010: Posters*, pages 1023–1031, Beijing, China. Coling 2010 Organizing Committee.

Rashmi Prasad, Susan Mcroy, Nadya Frid, Aravind Joshi, and Hong Yu. 2011. The biomedical discourse relation bank. *BMC bioinformatics*, 12:188.

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.

Magdaléna Rysová. 2012. Alternative lexicalizations of discourse connectives in Czech. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2800–2807, Istanbul, Turkey. European Language Resources Association (ELRA).

Magdaléna Rysová and Kateřina Rysová. 2018. Primary and secondary discourse connectives: Constraints and preferences. *Journal of Pragmatics*, 130:16–32.

Ted Sanders, Wilbert Spooren, and Leo G. M. Noordman. 1992. Toward a taxonomy of coherence relations. *Discourse Processes*, 15:1–35.

Wei Shi and Vera Demberg. 2017. On the need of 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.

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.

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.

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, Berlin, Germany. Association for Computational Linguistics.

Yue Yu, Yilun Zhu, Yang Liu, Yan Liu, Siyao Peng, Mackenzie Gong, and Amir Zeldes. 2019. GumDrop at the DISRPT2019 shared task: A model stacking approach to discourse unit segmentation and connective detection. In *Proceedings of the Workshop on Discourse Relation Parsing and Treebanking 2019*, pages 133–143, Minneapolis, MN. Association for Computational Linguistics.

Amir Zeldes, Debopam Das, Erick Galani Maziero, Juliano Antonio, and Mikel Iruskieta. 2019. The DISRPT 2019 shared task on elementary discourse unit segmentation and connective detection. In *Proceedings of the Workshop on Discourse Relation Parsing and Treebanking 2019*, pages 97–104, Minneapolis, MN. Association for Computational Linguistics.

Amir Zeldes and Yang Liu. 2020. A neural approach to discourse relation signal detection. *Dialogue Discourse*, 11(2):1–33.

Amir Zeldes, Yang Janet Liu, Mikel Iruskieta, Philippe Muller, Chloé Braud, and Sonia Badene. 2021. The DISRPT 2021 shared task on elementary discourse unit segmentation, connective detection, and relation classification. In *Proceedings of the 2nd Shared Task on Discourse Relation Parsing and Treebanking (DISRPT 2021)*, pages 1–12, Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Carbon Footprint

We access carbon footprints by using the *codecarbon*[1] framework. The corpus is processed once in the beginning and embeddings are persistently stored on disk for later usage. By preparing theses embeddings, we roughly emitted 7.93g carbon dioxide (CO2). Table 3 shows the carbon footprints for each model and provide averaged values per experiment. In total, as we avoid training large models by ourselves, and rather use features taken from pre-trained language models as provided, we emitted about 266.15g CO2 for all our experiments.

Model-Type	**Config**	**Emission (g CO2)**
candidate	advance ctx=0	2.97 (0.01)
candidate	advance ctx=1	3.05 (0.04)
candidate	advance ctx=2	2.98 (0.02)
candidate	simple ctx=0	3.96 (0.54)
candidate	simple ctx=1	3.99 (0.56)
candidate	simple ctx=2	4.42 (0.15)
label	ctx (all)	26.11 (3.09)
label	simple (all)	16.04 (0.49)
label	ctx (pos)	2.73 (0.61)
label	simple (pos)	1.79 (0.17)
sentence	ctx (all)	10.97 (0.93)
sentence	simple (all)	7.07 (0.54)

Table 3: Carbon footprint approximations averaged over runs with standard deviation.

B Extracted Candidate Patterns

The following tables give a more detailed overview about the pattern-extraction mechanisms. In Table 4, the top 30 extracted patterns are given for both approaches, exact extraction on the left side and context-sensitive extraction on the right side, with one additional token to the left and right. This is complemented in Table 5 by the number of extracted candidates per rule on a randomly sampled test set, as described in Section 5.

[1] https://github.com/mlco2/codecarbon

#	Exact	Count	Context-Sensitive	Count
1	VBG	468	, VBG DT	162
2	RB	113	, VBG NNS	39
3	VBN IN	59	, VBG PRP	35
4	IN DT NN	45	, VBG NN	31
5	DT VBZ	39	BOS IN DT NN ,	29
6	DT VBZ IN	31	, VBG JJ	26
7	IN RB	24	, VBG IN	25
8	IN NN IN	23	, VBG NNP	18
9	JJ	17	, VBN IN DT	18
10	VBG IN	16	BOS RB ,	17
11	DT NN VBZ	15	BOS IN RB ,	15
12	RB TO	15	BOS IN NN IN VBG	15
13	DT NN VBD	14	, VBG TO	14
14	IN VBD	14	, VBG JJR	14
15	RB VBG	12	, VBG PRP$	14
16	WP VBZ JJR	12	, RB TO VB	13
17	RB RB	12	BOS WP VBZ JJR ,	12
18	DT MD VB	12	, VBG IN DT	12
19	DT NN	11	, VBG RB	10
20	IN VBZ	11	BOS DT VBZ IN DT	10
21	IN DT	11	, VBN IN JJ	9
22	DT VBD	10	BOS DT NN :	8
23	IN NN	10	DT JJ NN	8
24	RB RB IN	10	, RB ,	7
25	DT NN VBD IN	9	BOS DT NN VBD DT	7
26	IN CD NN	9	BOS IN DT ,	7
27	VBD DT NN IN	9	BOS IN CD NN ,	7
28	DT JJ NN VBZ	8	, RB .	7
29	VBG RP	7	, VBG VBG	7
30	DT NN MD VB	7	BOS VBG DT	7

Table 4: Top 30 of extracted patterns from full data set. Comparison of simple patterns with their context-sensitive counter parts.

#	Exact	Count	Context-Sensitive	Count
1	NN	18686	JJ	1541
2	IN	13915	RB	1100
3	DT	10907	DT	552
4	JJ	7698	VBG	516
5	DT NN	5278	IN	245
6	IN DT	4654	DT NN	206
7	NN IN	4517	VBZ	142
8	RB	4509	VBN IN	128
9	VBD	3997	VBN	104
10	VB	3565	IN DT NN	76
11	VBN	2910	TO	73
12	VBZ	2623	DT NN VBD	58
13	IN DT NN	2133	VBG IN	54
14	VBG	1869	IN NN	51
15	TO	1761	RB RB	46
16	TO VB	1728	DT NN VBZ	45
17	IN NN	1590	PRP VBD	30
18	DT JJ NN	1418	DT VBZ	30
19	IN JJ	1182	IN DT JJ NN	29
20	MD VB	1061	JJR	28
21	VBN IN	1035	IN CD	26
22	IN DT JJ	996	VBD DT NN IN	25
23	IN CD	917	PRP VBZ	21
24	CD NN	871	IN NN IN	20
25	IN NNS	765	VBD	17
26	NN VBZ	728	DT NN MD VB	15
27	NNS VBP	648	RB JJ	13
28	PRP VBD	626	VBG DT NN	13
29	IN DT JJ NN	622	RB TO	12
30	VBD IN	557	IN NNS	12

Table 5: Top 30 of extracted patterns from randomly sampled test partition. The ordering differs between **Exact** and **Context-Sensitive** due to different patterns. In comparison, context-sensitive patterns are extract more different, but much less candidates per pattern, for example interjections (IN) on the left are extracted about 57 times more than on the right.