A Pilot Study on Dialogue-Level Dependency Parsing for Chinese

Gongyao Jiang[1], Shuang Liu[2], Meishan Zhang[3]*, Min Zhang[3]
[1]School of New Media and Communication, Tianjin University, China
[2]College of Intelligence and Computing, Tianjin University, China
[3]Institute of Computing and Intelligence, Harbin Institute of Technology (Shenzhen), China
jianggongyao@gmail.com, shuang.liu@tju.edu.cn
mason.zms@gmail.com, zhangmin2021@hit.edu.cn

Abstract

Dialogue-level dependency parsing has received insufficient attention, especially for Chinese. To this end, we draw on ideas from syntactic dependency and rhetorical structure theory (RST), developing a high-quality human-annotated corpus, which contains 850 dialogues and 199,803 dependencies. Considering that such tasks suffer from high annotation costs, we investigate zero-shot and few-shot scenarios. Based on an existing syntactic treebank, we adopt a signal-based method to transform seen syntactic dependencies into unseen ones between elementary discourse units (EDUs), where the signals are detected by masked language modeling. Besides, we apply single-view and multi-view data selection to access reliable pseudo-labeled instances. Experimental results show the effectiveness of these baselines. Moreover, we discuss several crucial points about our dataset and approach.

1 Introduction

As a fundamental topic in the natural language processing (NLP) community, dependency parsing has drawn a great deal of research interest for decades (Marcus et al., 1994; McDonald et al., 2013; Zhang et al., 2021). The goal of dependency parsing is to find the head for each word and the corresponding relation (Kübler et al., 2009). Most of previous works have focused on the sentence level, while the dialogue-level dependency parsing still stands with the paucity of investigation.

Prior studies build dialogue-level discourse parsing datasets (Asher et al., 2016; Li et al., 2020) with reference to the text-level discourse dependency (Li et al., 2014). The discourse structures in these data are constructed by elementary discourse units (EDUs) and the relationships between them, without regard to the inner structure of EDUs. It is of interest to incorporate both inner-EDU and inter-EDU dependencies throughout a dialogue to construct a word-wise dependency tree, which is in line with sentence-level dependencies and further able to express hierarchical structures. Hence, we form the dialogue-level dependency, by adapting commonly-used syntactic dependency (Jiang et al., 2018) and rhetorical structure theory (RST) dependency (Carlson et al., 2001; Li et al., 2014) into inner-EDU and inter-EDU dependencies.

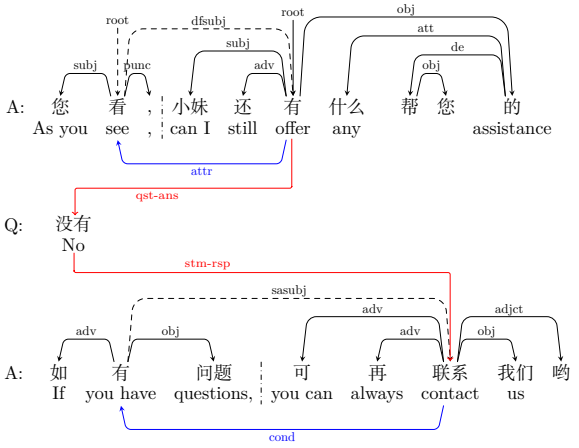

Figure 1: A fragment example of dialogue-level dependencies. Vertical dashed lines are separation boundaries of EDUs. Above words are the inner-EDU dependency arcs, while the arcs below words or cross utterances represent the inter-EDU dependencies.

The scarcity of research on dialogue-level dependency parsing might be caused by the prohibitive cost of annotation. Thus, we focus on low-resource settings, aiming to construct a sufficient test set to support reliable evaluation and label a small training set. We craft an annotation guideline and develop a platform for human labeling. After that, we perform manual annotation and obtain 50 training instances and 800 test samples. Figure 1 illustrates a fragment in a labeled dialogue, which contains three utterances with dependency links.

Learning a parser with scant to no supervised signals can be challenging. Fortunately, it is straight-

*Corresponding author

Findings of the Association for Computational Linguistics: ACL 2023, pages 9526–9541
July 9-14, 2023

forward to use an existing syntactic treebank to attain inner-EDU dependencies. Meanwhile, we find some overlap between syntactic dependency and inter-EDU dependency. As shown in Figure 1, "offer" is dependent on "see" with the "dfsubj" (different subject) relationship in syntactic dependency, which matches the "attr" (attribution) of inter-EDU dependency. Furthermore, inter-EDU arcs between utterances are often emitted from the root node above to the root below. Hence, we can naturally obtain inter-EDU arcs from partial syntactic dependencies.

We find that certain words can reflect the discourse role of EDUs, helping to assign labels to inter-EDU arcs. For instance, "see" and "if" reflect the "attribution" and "condition" signals respectively, illustrated in Figure 1. Thus, we propose a method to discover and leverage these signals for inter-EDU label assignment. Inspired by prompt-based learning (Liu et al., 2021), we adopt masked language modeling (MLM) to recover signal words, the set of which is carefully defined in advance. Predicted words are then mapped to corresponding signals. Based on these signals, a handful of simple well-defined rules can infer credible dependencies.

The gap between the syntactic treebank and dialogue-level dependencies goes beyond labels; differences in text style expose a trained parser to the out-of-distribution (OOD) effect (Li et al., 2019). To alleviate that, we utilize unlabeled dialogue data to bridge the cross-domain gap. We borrow ideas from self-training (Scudder, 1965) and co-training (Blum and Mitchell, 1998), applying single-view and multi-view data filtering augmentation. Given pseudo-labeled samples predicted by parsers, we calculate their confidence scores and set a threshold. Instances above the threshold are provided as additional training samples.

We carry out experiments in zero-shot and few-shot scenarios. The empirical results suggest that our signal-based baseline can achieve reasonable performance in the zero-shot condition and provide significant improvement in the few-shot setting. Incorporating filtered pseudo-labeled data further advances performance. In addition, we present several discussion topics, which can affirm directive intuitions and crucial practices in this work.

Our contributions are mainly in three respects:

- We build the first Chinese dialogue-level dependency treebank.
- We propose signal-based dependency transformation and pseudo-labeled data filtering approaches as parsing baselines.
- The proposed methods achieve reasonable parsing performance in low-resource settings.

All our datasets and codes will be publicly available at github.com/Zzoay/DialogDep for research purpose.

2 Dataset Construction

To facilitate our research on dialogue-level dependency parsing, here we construct a high-quality corpus manually, named the Chinese dialogue-level dependency treebank (CDDT). The treebank borrows the sentence-level dependency (Jiang et al., 2018) and the discourse-level dependency (Li et al., 2014)[1], extending both to the dialogue texts. Below, we present the annotation details.

2.1 Annotation Preparation

Data Collection. We use a publicly available dialogue dataset (Chen et al., 2020) as our raw corpus, which is collected from real-world customer service, containing multiple turns of utterance. We access this data from an available source.[2] We use 800 data from the full test set as the raw corpus of our test set and randomly sample 50 data from the 9101 training data as our raw corpus for few-shot learning. The data providers have segmented words and anonymized private information. We further manually clean the data, removing noise text and performing fine-grained word segmentation.

Guideline. A well-defined dialogue-level dependency tree can be decomposed into two parts, i.e., inner-EDU dependencies and inter-EDU dependencies, respectively. Following Carlson et al. (2001), we treat EDUs as non-overlapping spans of text and use lexical and syntactic clues to help determine boundaries. To adapt to dialogue parsing and linguistic features of Chinese, we simplify the boundary determination, regarding the boundaries as the leftmost and rightmost words of spans, whose words are covered by a complete syntactic subtree. The root nodes of each subtree are often predicates that reflect single semantic events, illustrated in Figure 1.

For the inner-EDU dependency annotation, we follow the guideline proposed by Jiang et al. (2018),

[1] Appendix A provides a brief introduction to this discourse-level parsing schema, i.e., RST.

[2] As the public link to this data is inaccessible, we utilize another version made public by CSDS (Lin et al., 2021), a dialogue summary dataset, based on the same dialogue data.

Statistic	Train	Test
# dialogue	50	800
avg.# turns	23	25
avg.# words	194	212
# inner	9129	159803
# inter	1671	29200

Table 1: Some statistical information about CDDT. "#" and "avg.#" represent "the number of" and "the average number of" respectively.

which includes 21 classes of syntactic dependency. For the inter-EDU one, we examine and adapt the EDU-wise relations (Carlson et al., 2001) to word-wise dependencies, including 15 coarse-grained relations and 4 fine-grained relations divided from "topic-comment" to accommodate connections between utterances. In annotation, inner-EDU dependencies should be annotated first as the underlying basis, while inter-EDU dependencies annotation should take full account of the semantic relations between EDUs. Appendix B shows all the dependencies and reveals more details about our dialogue-level dependencies.

2.2 Human Annotation

Platform. The dialogue-level dependency annotation consists of inner-utterance and inter-utterance parts, where the former is in line with the common sentence-level annotation, and the latter needs to connect from above to below. To accommodate this form of human labeling, we develop an online annotation platform. Annotators can easily label the inner-utterance and inter-utterance dependencies on this platform by linking tags between words.

Annotation Process. We employ two postgraduate students as annotators, both of whom are native Chinese speakers. Each annotator is intensively trained to familiarise the guideline. To guarantee the labeling quality, we apply a double annotation and expert review process. Each annotator is assigned to annotate the same raw dialogue. The annotation is accepted if the two submissions are the same. Otherwise, an experienced expert with a linguistic background will compare the two submissions and decide on one answer. Moreover, the expert checks all labeling for quality assurance.

Statistics. In the end, we arrive at 800 number of annotated dialogues for testing and 50 for few-shot training. Table 1 shows some statistical information. It can be seen that the number of dialogue rounds is large, leading to an expensive labeling process. Also, inter-EDU relations are much more scarce compared to inner-EDU ones, as inter-EDU dependencies are higher-level relations that reflect the connections between EDUs. Appendix B shows the quantities of each category.

3 Methodology

Dialogue-level dependency parsing involves two folds, namely inner-EDU dependency parsing and inter-EDU dependency parsing. We leverage an existing syntactic treebank $\mathcal{S}$ (Li et al., 2019) to learn a parser, which can analyze inner-EDU dependencies well. Meanwhile, it can be observed that syntactic dependencies have the potential to be converted into inter-EDU ones (Figure 1). Thus, we propose a signal-based method to perform dependency transformation for parser training and inference. Moreover, we apply pseudo-labeled data filtering to leverage the large-scale unlabeled dialogue data $\mathcal{D}$.

3.1 Signal-based Dependency Transformation

Our dependency parsing model contains an encoder layer that uses a pretrained language model (PLM) and a biaffine attention decoder as detailed in Dozat and Manning (2017). Given a sentence (in $\mathcal{S}$) or utterance (in dialogues) $\boldsymbol{x} = [w_1, w_2, \ldots, w_n]$, the parser can output the arc and label predictions:

$$\bar{\boldsymbol{y}}^{(arc)}, \bar{\boldsymbol{y}}^{(label)} = \text{Parser}\,(\boldsymbol{x}) \quad (1)$$

In summary, the dependency transformation can be applied in two ways. The first is replacing partial labels of predicted syntactic dependencies with inter-EDU ones, similar to post-processing. We denote it to *PostTran*. The second is to transform the syntactic dependency dataset $\mathcal{S}$ into the one with extended inter-EDU dependencies for parser training, denoted to *PreTran*. For clarity, we denote the transformed treebank to $\mathcal{S}_\tau$. We refer the parser trained on the syntactic treebank $\mathcal{S}$ to Parser-S, and the parser based on $\mathcal{S}_\tau$ to Parser-T.

EDU Segmentation. When *PreTran*, a sentence in $\mathcal{S}$ is firstly separated into several EDUs. We exploit a simple but effective way, treating punctuation such as commas, full stops, and question marks as boundaries between EDUs.[3] Meanwhile, some specific syntactic labels (e.g. sasubj, dfsubj) that span

[3] In our preliminary experiments, this method achieves a 68.82% F1 score of segmentation in those utterances with multiple EDUs, while the single EDU predictions' F1 can be achieved to 90.10%. We show all the segmentation operators in the publicly available code.

a range can also be considered as implicit EDU segmenting signals (Figure 1). When *PostTran*, an utterance in dialogue is segmented in the same way. Additionally, there are clear boundaries between utterances, which can be used to separate EDUs.

Given the segmented EDUs, the next issue is where the inter-EDU arcs fall and in what direction. We find that inside utterances, inter-EDU dependencies are often overlapped with certain syntactic dependencies (Figure 1). We name the labels of these syntactic labels as "transforming labels" and predefine a set of them $L = \{root, sasubj, dfsubj\}$. If $\boldsymbol{y}_i^{(label)} \in L$, we can directly retain (sometimes reverse) the arcs and convert the labels to inter-EDU ones. Besides, there is no predicted relationship between utterances. We find that most inter-utterance arcs are emitted from the root node above to the root node below. Thus, we link the predicted roots from above to below as the inter-EDU arcs between utterances.

MLM-based Signal Detection. Given those inter-EDU arcs, we propose a signal-based transform approach to assign inter-EDU labels to them. First, we introduce the signal detection method, which is based on a masked language modeling (MLM) paradigm. We find that certain words reflect the semantic role of EDUs. For instance, an EDU that contains "if" is commonly the "condition" role (Figure 1). We can emit an arc "cond" from its head EDU to it. Thus, we pre-define a word-signal dictionary, mapping words to the corresponding inter-EDU relation signals. A word that reflects the signal is called "signal word".

Subsequently, we apply the MLM on the large-scale unlabeled dialogue data $\mathcal{D}$ to learn inter-EDU signals. During the training stage, the signal word v is randomly dropped in a segmented EDU $\boldsymbol{e}$, and a masked language model is to recover it. Like prompt-based learning (Liu et al., 2021), we modify $\boldsymbol{e}$ by a slotted template into a prompt $\boldsymbol{e'}$. The slot is a placeholder with "[mask]" tokens. Next, a model with PLM and MLP decoder outputs the word distribution in masked positions,[4] as distribution of signal words:

$$P\left(v|\boldsymbol{e}\right) = \text{softmax}\left(\text{MLP}\left(\text{PLM}\left(\boldsymbol{e'}\right)\right)\right) \quad (2)$$

where the MLP is to project the hidden vector to vocabulary space.

[4]Since signal words may contain multiple characters in Chinese, the masked place should span several [mask] tokens. We average the output probabilities. The implemented details are offered in Appendix C.

At the inference stage, the model outputs the distribution of signal words. The probabilities of signal words are grouped and averaged by their corresponding signals. The grouped probabilities form the distribution of inter-EDU signals.

$$P\left(s|\boldsymbol{e}\right) = \text{GroupMean}\left(P\left(v|\boldsymbol{e}\right)\right) \quad (3)$$

The signal s can be obtained by argmax $P\left(s|\boldsymbol{e}\right)$. We expand the predicted signal to the whole $\boldsymbol{e}$. By executing the above procedure in batch with $\boldsymbol{e}$ in $\boldsymbol{x}$, we end up with the signal sequence, which is denoted in bold $\boldsymbol{s}$.

Signal-based Transformation. Given detected signals $\boldsymbol{s}$, Algorithm 1 show how partial syntactic labels are transformed into inter-EDU labels. We pre-set 3 conditions and the corresponding strategies to cover the majority of cases. 1) If the head node of the current word crosses the EDU's boundary, or the label $\boldsymbol{y}_i^{(label)} \in L$ and the connection spans k, we assign the label by the corresponding signal. 2) If $\boldsymbol{s}_i$ is "cond" or "attr", we reverse the connections between head and tail nodes. 3) If greetings are the root EDU, then we invert the links, since we are more interested in the core semantics. The *FindTail* function in Algorithm 1 performs an iterative procedure. It looks for the first node in other EDUs whose head node has an index equal to the current index i. Meanwhile, the labels of arcs between utterances are directly assigned by the signals of their head words.

3.2 Unlabeled Data Utilization

We leverage large-scale dialogue data $\mathcal{D}$ to bridge the distribution gap between the syntactic treebank and our annotated corpus. It is straightforward to use a well-trained parser to assign pseudo labels to $\mathcal{D}$ for subsequent training. Nevertheless, this pseudo-labeled data includes a sea of mislabels, which in turn jeopardize performance. Thus, we apply single-view and multi-view methods to select high-quality additional samples.

Single-view. In intuition, predicted samples with higher confidence are of higher quality. Given an unlabeled utterance $\boldsymbol{x}$, a well-trained parser (Parser-S or Parser-T) can output the probabilities of each arc and label, without the final decoding:

$$P\left(\boldsymbol{y}^{(arc)}\middle|\boldsymbol{x}\right), P\left(\boldsymbol{y}^{(label)}\middle|\boldsymbol{x}\right) = \text{Parser}^{*}\left(\boldsymbol{x}\right) \quad (4)$$

Then, we average the highest probabilities in each position, as the confidence of one utterance.

Algorithm 1 Signal-based Transformation

Require: a sentence (utterance) $\boldsymbol{x}$; arcs $\boldsymbol{y}^{(arc)}$; labels $\boldsymbol{y}^{(label)}$; inter-EDU signals $\boldsymbol{s}$; set of transforming labels L; sequence length n.

for $i = 1$ to n **do**
  if condition 1 **then** ▷ certain labels
    $\boldsymbol{y}_i^{(label)} \leftarrow \boldsymbol{s}_i$
    if condition 2 **then** ▷ special cases
      $t \leftarrow FindTail(\boldsymbol{y}_i^{arc})$
      $\boldsymbol{y}_t^{(arc)} \leftarrow \boldsymbol{y}_i^{(arc)}$
      $\boldsymbol{y}_i^{(arc)} \leftarrow t$
    else if condition 3 **then** ▷ greetings
      $t \leftarrow FindTail(\boldsymbol{y}_i^{(arc)})$
      $\boldsymbol{y}_i^{(arc)}, \boldsymbol{y}_i^{(label)} \leftarrow t$, "elbr"
      $\boldsymbol{y}_t^{(arc)}, \boldsymbol{y}_t^{(label)} \leftarrow 0$, "root"
    end if
  end if
end for
return $\boldsymbol{y}^{(arc)}, \boldsymbol{y}^{(label)}$

Equation 5 shows the calculation of arc confidence $c^{(arc)}$. The label confidence $c^{(label)}$ is computed in the same way.

$$c^{(arc)} = \frac{1}{n}\sum_{i}^{n} \max P\left(\boldsymbol{y}_i^{(arc)} \middle| \boldsymbol{x_i}\right) \quad (5)$$

A pseudo-labeled utterance is reserved when its $c^{(arc)}$ and $c^{(label)}$ both greater than a confidence threshold ϵ. The filtered samples are incorporated with $\mathcal{S}$ or $\mathcal{S}_\tau$ for training a new parser.

Multi-view. Prior research (Blum and Mitchell, 1998) and our pilot experiment suggest that multi-view productions of pseudo data outperform single-view ones. Thus, we exploit Parser-S and Parser-T to conduct multi-view data augmentation.

The confidence computation and filtering methods are the same as the single-view ones above. The filtered samples labeled by Parser-T and Parser-S are merged together and de-duplicated by their confidence magnitude (i.e. if two samples are the same but have different labels, the one with higher confidence will be retained). Then, the merged instances are added to $\mathcal{S}$ or $\mathcal{S}_\tau$ for subsequent training. In this way, parsers trained on the set with such data augmentation can access complementary information.

4 Experiment

4.1 Settings

Evaluation. We use the labeled attachment score (LAS) and the unlabeled attachment score (UAS) for evaluation. For a fine-grained analysis, we report the scores of inner-EDU and inter-EDU dependencies. In the absence of the development set in our low-resource scenarios, we retain the last training checkpoint for evaluation. To maintain a balance between energy savings and results reliability, in zero-shot settings, we set a random seed of 42 for all experiments and report the testing results. In few-shot settings, we repeat data sampling on 5 random seeds in 4 few-shots (5, 10, 20, 50) settings. Then we choose the seeds that obtain median scores for training and final evaluation. All experiments are carried out on a single GPU of RTX 2080 Ti.

Hyper-parameters. Our PLM is a Chinese version of ELECTRA (Clark et al., 2020), implemented by Cui et al. (2020). We exploit the base scale discriminator[5] for fine-tuning. The hidden size of the subsequent parts of our Parser and MLM is set to 400, and the dropout is 0.2. We train our models by using the AdamW optimizer (Loshchilov and Hutter, 2017), setting the initial learning rate of the PLM to 2e-5 and of the subsequent modules to 1e-4, with a linear warmup for the first 10% training steps. The weight decay is 0.01. We apply the gradient clipping mechanism by a maximum value of 2.0 to alleviate gradient explosion. The training batch size is 32 and the epoch number is 15. We set the minimum span k of connections to 2. Moreover, we set the iteration number of pseudo-labeled data selection to 1. The confidence threshold ϵ is 0.98.

4.2 Results

The approaches in this work can be thought of as a permutation of data, parsers, and filtering methods. There are two sets of data, the syntactic dependency treebank $\mathcal{S}$ ($\mathcal{S}_\tau$), and the large-scale unlabeled dialogue data $\mathcal{D}$. Also, there are two types of transformations *PreTran* and *PostTran*.[6] For brevity, we simplify the Parser-S to f_s and the Parser-T to f_t. The pseudo-labeled data selection is denoted by a function η. We record and analyze the experimental results of zero-shot and few-shot settings.

[5] huggingface.co/hfl/chinese-electra-180g-base-discriminator

[6] Predictions for all methods can be processed by *PostTran*. For consistency and convenience, our reported results are with *PostTran*. Appendix D shows more details.

Training Data	Inner	Inter
$\mathcal{S}$	83.14	49.94
$\mathcal{S}_\tau$	83.16	49.71
$f_s(\mathcal{D})$	82.13	48.76
$f_t(\mathcal{D})$	82.14	48.38
$(f_s + f_t)(\mathcal{D})$	82.46	49.03
$\mathcal{S} + f_s(\mathcal{D})$	82.48	48.84
$\mathcal{S}_\tau + f_t(\mathcal{D})$	82.68	49.02
$\mathcal{S} + \eta(f_s(\mathcal{D}))$	84.14	50.47
$\mathcal{S}_\tau + \eta(f_t(\mathcal{D}))$	84.13	50.27
$\mathcal{S} + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$	84.24	50.62
$\mathcal{S}_\tau + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$	**84.34**	**50.78**

Table 2: The test results under the zero-shot setting. We divide the results by a dashed line according to the source of training data. "Inner" represents the inner-EDU dependency, and "Inter" is the inter-EDU one.

Zero-shot. Since the trends in the UAS and LAS are roughly consistent, here we only report the LAS. Details are recorded in Appendix D. As shown in Table 2, the parser trained on syntactic treebank $\mathcal{S}$ and transformed treebank $\mathcal{S}_\tau$ achieve similar performances in inner-EDU parsing. This shows how little our dependency transforming method disrupts the original syntactic structure. It is intuitive that the parser independently trained on pseudo-labeled dialogue data $\mathcal{D}$ performs not well. The parsers trained on the mergers of $\mathcal{S}$ and $\eta(\mathcal{D})$ obtain the highest syntax parsing scores, demonstrating the usefulness of our data selection.

For the inter-EDU dependency parsing, it can be observed that the transformed treebank $\mathcal{S}_\tau$ makes the parser perform better. Similar to inner-EDU parsing, the direct use of $\mathcal{D}$ performs poorly. Although the ensemble of Parser-S and Parser-T can bring some performance gains, they are limited. The combination of $\mathcal{S}$ ($\mathcal{S}_\tau$) and $\eta(\mathcal{D})$ is useful. The parser trained on $\mathcal{S}_\tau + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$ achieves best inner-utterance performance. It demonstrates the effectiveness of our signal-based dependency transformation and multi-view data selection.

Few-shot. We conduct few-shot (5, 10, 20, 50) experiments, incorporating the few labeled samples with those typologies of training data in the above zero-shot setting. For clarity, we also present only the LAS. Appendix E gives more details.

As can be observed in Table 3, our approaches significantly outperform parsers trained on only a small number of labeled training samples. The performance improvement of inter-EDU parsing is more pronounced than the one of inner-EDU parsing, probably thanks to the fact that the former is a more difficult task with larger room for enhancement. It can also be observed that as the annotated training data increases, the performance of inner-EDU parsing gradually reaches a higher level, and the improvement achieved by introducing augmented data becomes confined. Nevertheless, parsers trained on the data with augmentation $\mathcal{S}_\tau + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$ achieves highest scores in the majority of few-shot settings, showing the effectiveness of the proposed method.

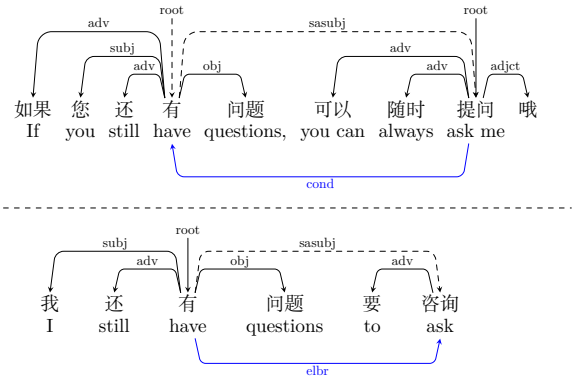

Figure 2: Two examples, where the above has a different rhetorical structure than the below, while they are almost isomorphic based on syntactic dependencies.

5 Discussion

What are the advantages of introducing discourse dependency compared to only syntax? Figure 1 has illustrated the superiority of introducing discourse dependency in two aspects. Relationships between utterances can be expressed by discourse dependency. Also, it represents a hierarchical structure and enables a more nuanced connection between EDUs.

Figure 2 gives another sample, the syntactic dependency suffers from the problem of somorphism. Those two sentences contain different semantic information, while their two syntactic subtrees are simply connected by "sasubj", resulting in their highly analogous architecture. According to RST dependency, the two subtrees of the first sentence are linked by "cond" (condition) from right to left, and the two below are connected by "elbr" (elaboration) from left to right. The inclusion of discourse dependency can alleviate the dilemma of somorphism by expressing high-level hierarchies.

How much overlap between partial syntactic dependencies and inter-EDU ones? Figure 1 pro-

Augmented data	5		10		20		50	
	Inner	**Inter**	**Inner**	**Inter**	**Inner**	**Inter**	**Inner**	**Inter**
Null	40.77	25.61	55.83	30.98	70.44	42.30	82.64	51.00
$\mathcal{S}$	85.01	50.02	85.83	50.98	86.80	52.04	88.20	54.42
$\mathcal{S}_\tau$	85.05	50.09	85.69	51.13	86.77	52.01	**88.28**	54.53
$\mathcal{S} + \eta\left(f_s\left(\mathcal{D}\right)\right)$	85.43	50.58	85.70	51.56	**87.03**	52.51	88.18	55.50
$\mathcal{S}_\tau + \eta\left(f_t\left(\mathcal{D}\right)\right)$	85.22	50.78	85.81	51.61	86.75	52.69	88.12	55.61
$\mathcal{S}_\tau + \eta\left(f_s\left(\mathcal{D}\right) + f_t\left(\mathcal{D}\right)\right)$	**85.55**	**51.12**	**85.88**	**51.88**	86.92	**52.86**	88.20	**55.73**

Table 3: The test results under the few-shot settings. Here we report the LAS.

vides an example that demonstrates the potential of transforming syntactic dependencies into inter-EDU ones. Here we present a quantitative analysis.

We present a matching score to measure the extent of overlap. We first obtain syntactic predictions by Parser-S. Then we replace those dependencies that bear a specific syntactic label with an inter-EDU class. Next, we compute the LAS for that class, as the matching score between the syntactic and inter-EDU labels. Figure 3 illustrates the top-5 matching scores of syntactic labels "root", "sasubj", and "dfsubj". It can be seen that these syntactic labels have a respectable consistency with certain inter-EDU classes, revealing the potential for dependency conversion. Interestingly, different syntactic labels are matched with different inter-EDU classes, indicating the overlap of specific syntactic structures with certain discourse structures.

How broadly can inter-EDU signals be reflected by pre-defined signal words? As mentioned above, certain words can reflect the semantic role of EDU. We calculate the consistency of signal words and the relationship labels on EDUs to quantify the extent. Given an EDU, we directly assign its label by the signal corresponding to the pre-defined signal word it contains. Then, we compute the accuracy of each inter-EDU label as the matching score of the relevant signals. It can be observed in Figure 4 that some labels such as "elbr" (27), "qst-ans" (37), and "stm-rsp" (38) can be strongly reflected by the pre-defined signal words. Some labels that do not contain significant signals in EDUs are maintained at low scores, especially "bckg" (22) and "topic-chg" (35), indicating that there is much room for improvement.

Is the MLM-based signal detection method able to detect implicit signal? Our inter-EDU signal detection method applies an MLM paradigm, which tries to recover the dropped signal word during the training stage. Intuitively, this method should have the capacity to predict those implicit signals. Figure 5 gives two examples to prove that. It can be seen that even when "if" is removed, the approach of MLM-based signal detection can still predict the "cond" signal, which is denoted as a 25 number. Based on this, the parser can correctly predict the "cond" relations between two EDUs.

How does the reserved size of pseudo-labeled data change by confidence thresholds? We vary the threshold ϵ from 0.5 to 0.98 to investigate the changes in the amount of data selected. Figure 6 illustrates the trend. Interestingly, the size of selected data is still large even though the threshold is greater than 0.9, illustrating the high confidence in dependency parsing. It can also be observed that the increase in merged data becomes more pronounced as the threshold increases. Intuitively, as the amount of data selected decreases, fewer data will overlap.

Why not leverage Chinese RST treebanks to train a parser? To date, there exist two Chinese RST treebanks, the RST Spanish-Chinese Treebank (Cao et al., 2018) and GCDT (Peng et al., 2022). It is natural to use them to provide supervised signals for inter-EDU parsing. In our prior experiments, we find that directly using these treebanks for training a parser leads to unsatisfactory performance. One possible reason is that their annotation style is not aligned with ours. For instance, GCDT splits sentences very finely and uses lots of "same-unit" relations to align to the English style, fitting their multilingual tasks. This occurs not in our data as it tends to cause an incomplete structure within the EDU. Furthermore, GCDT lacks some of the label categories in our data, especially some common relations such as "statement-response". In addition, inconsistencies in data distribution can also contribute to underperformance.

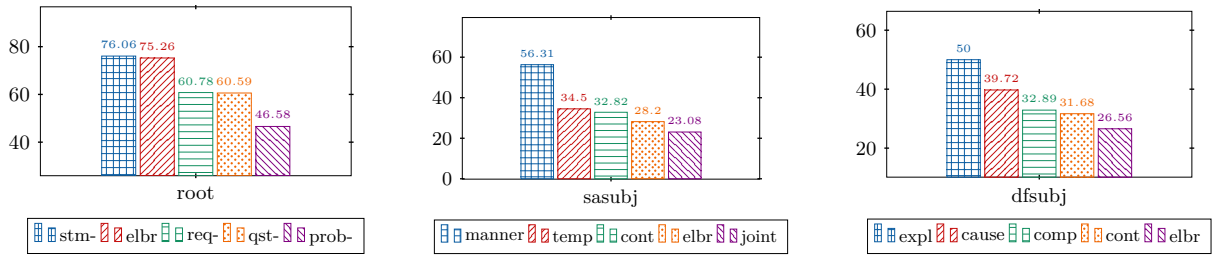

Figure 3: Top-5 matching labels and scores of "root", "sasubj", and "dfsubj".

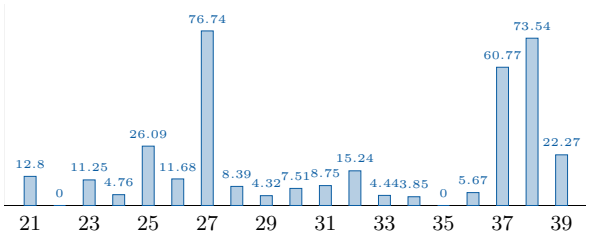

Figure 4: Matching score of signals. The numbers below sentences indicate the signals detected, in the order given in Appendix B.

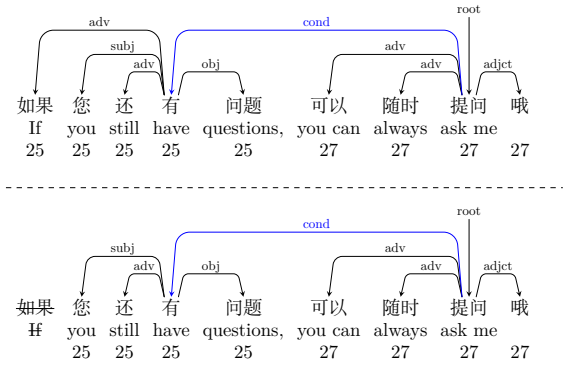

Figure 5: Two cases, where the above is with the explicit signal word "If" and the below is without it. The numbers represent the labels in the corresponding order.

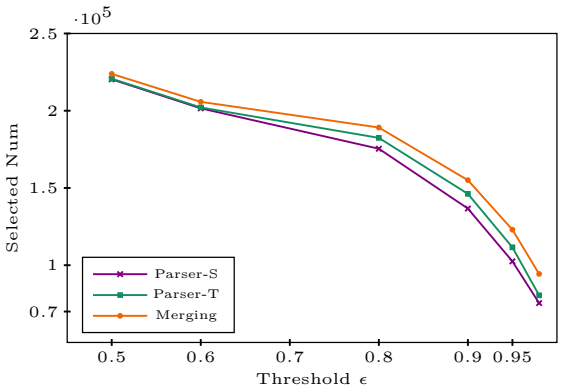

Figure 6: Quantity variation of filtered instances.

6 Related Work

Dependency parsing. To date, there are several Chinese dependency paradigms and the corresponding treebanks (Xue et al., 2005; Che et al., 2012; McDonald et al., 2013; Qiu et al., 2014). These works mostly focus on sentence-level dependency parsing, while the document-level one is conspicuous by its paucity. Li et al. (2014) adopt a dependency parsing paradigm for discourse parsing, while its EDU-wise pattern neglects to parse inside EDUs. We propose a unified schema, which includes both word-wise dependencies within and between EDUs that organize whole dialogues into tree structures.

Dialogue parsing. Discourse structures can be expressed by several theories, e.g., RST (Mann and Thompson, 1987, 1988), SDRT (Asher and Lascarides, 2003), and PTDB (Prasad et al., 2008). The investigation of dialogue-level discourse parsing is still in its early stage. Afantenos et al. (2015); Asher et al. (2016) build an annotated corpus STAC based on SDRT (Asher and Lascarides, 2003), using a dependency paradigm for annotation of relationships between EDUs. Li et al. (2020) present the Molweni dataset, contributing discourse dependency annotations. These two data are annotated at the granularity of EDUs and, to accommodate multi-party dialogue scenarios, their annotations are based on SDRT that fits the graph structure. Differently, in line with common syntactic dependencies, the inter-EDU part of our dialogue-level dependency is word-wise and references RST (Carlson et al., 2001), which organizes a text into the tree structure. Furthermore, the prevalent corpus for dialogue dependency parsing is in English, and our dataset fills a lacuna in the Chinese corpus.

Weakly supervised learning. It is challenging to predict new classes that are unobserved beforehand (Norouzi et al., 2014). PLMs (Devlin et al., 2019; Clark et al., 2020; Brown et al., 2020) can address this challenge through language model-

ing combined with label mapping, and prompt-based learning can stimulate this ability (Liu et al., 2021; Ouyang et al., 2022; Lang et al., 2022). Inspired by that, we adopt an MLM-based method to sense inter-EDU signals and map them to unseen dependencies. Furthermore, our approach employs single-view and multi-view data selection, borrowing from self-training (Scudder, 1965) and co-training (Blum and Mitchell, 1998), which are used for growing a confidently-labeled training set.

7 Conclusion

We presented the first study of Chinese dialogue-level dependency parsing. First, we built a high-quality treebank named CDDT, adapting syntactic (Jiang et al., 2018) and RST (Carlson et al., 2001) dependencies into inner-EDU and inter-EDU ones. Then, we conducted manual annotation and reached 850 labeled dialogues, 50 for training and 800 for testing. To study low-resource regimes, we leverage a syntactic treebank to get inner-EDU dependencies and induce inter-EDU ones. We employed an MLM method to detect the inter-EDU signals of each EDU and then assign the detected signals to the arcs between EDUs. Furthermore, we exploited single-view and multi-view approaches for pseudo-labeled sample filtering. Empirical results suggest that our signal-based method can achieve respectable performance in zero-shot and few-shot settings, and pseudo-labeled data utilization can provide further improvement.

Limitations

This study suffers from four limitations. The first limitation is that even though we annotated 850 dialogues manually, which includes almost 200,000 dependencies, there is still room for improvement in the total number of labeled dialogues. The second one is that our parsing method of inter-EDU in the inter-utterance situation is simplistic and straightforward, and it can not cover certain difficult labels. It is desirable to propose a more elegant and comprehensive approach. The third is somewhat analogous to the second. In the future, we should propose an end-to-end method that replaces the current approach, which consists of several processing steps. The last one is about our pseudo-labeled data selection method. It could be interesting to investigate the iterative process.

Ethics Statement

We build a dialogue-level dependency parsing corpus by crowd annotations. The raw dialogue data is obtained from an open source. Besides, we remove information relating to user privacy. The annotation platform is developed independently by us. All annotators were properly paid by their efforts. This dataset can be employed for dialogue-level dependency parsing in both zero-shot and few-shot setting as well as in any other data settings.

Acknowledgement

We would like to thank the anonymous reviewers for their constructive comments, which help to improve this work. This research is supported by the National Natural Science Foundation of China (No. 62176180).

References

Stergos Afantenos, Eric Kow, Nicholas Asher, and Jérémy Perret. 2015. Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 928–937, Lisbon, Portugal. Association for Computational Linguistics.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Nicholas Asher and Alex Lascarides. 2003. *Logics of conversation*. Cambridge University Press.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Shuyuan Cao, Iria da Cunha, and Mikel Iruskieta. 2018. The RST Spanish-Chinese treebank. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 156–166, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Lynn Carlson and Daniel Marcu. 2001. Discourse tagging reference manual. *ISI Technical Report ISI-TR-545*, 54(2001):56.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurovsky. 2001. Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*.

Wanxiang Che, Zhenghua Li, and Ting Liu. 2012. Chinese dependency treebank 1.0 ldc2012t05. *Philadelphia: Linguistic Data Consortium*.

Meng Chen, Ruixue Liu, Lei Shen, Shaozu Yuan, Jingyan Zhou, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020. The JDDC corpus: A large-scale multi-turn Chinese dialogue dataset for E-commerce customer service. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 459–466, Marseille, France. European Language Resources Association.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 657–668, Online. 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.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Xinzhou Jiang, Zhenghua Li, Bo Zhang, Min Zhang, Sheng Li, and Luo Si. 2018. Supervised treebank conversion: Data and approaches. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2706–2716, Melbourne, Australia. Association for Computational Linguistics.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis lectures on human language technologies*, 1(1):1–127.

Hunter Lang, Monica N Agrawal, Yoon Kim, and David Sontag. 2022. Co-training improves prompt-based learning for large language models. In *International Conference on Machine Learning*, pages 11985–12003. PMLR.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2642–2652, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sujian Li, Liang Wang, Ziqiang Cao, and Wenjie Li. 2014. Text-level discourse dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25–35, Baltimore, Maryland. Association for Computational Linguistics.

Zhenghua Li, Xue Peng, Min Zhang, Rui Wang, and Luo Si. 2019. Semi-supervised domain adaptation for dependency parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395, Florence, Italy. Association for Computational Linguistics.

Haitao Lin, Liqun Ma, Junnan Zhu, Lu Xiang, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2021. CSDS: A fine-grained Chinese dataset for customer service dialogue summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4436–4451, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *CoRR*, abs/2107.13586.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

William C Mann and Sandra A Thompson. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal Dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S. Corrado, and Jeffrey Dean. 2014. Zero-shot learning by convex combination of semantic embeddings. *International Conference on Learning Representations*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Siyao Peng, Yang Janet Liu, and Amir Zeldes. 2022. GCDT: A Chinese RST treebank for multigenre and multilingual discourse parsing. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pages 382–391, Online only. 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 *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Likun Qiu, Yue Zhang, Peng Jin, and Houfeng Wang. 2014. Multi-view chinese treebanking. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 257–268.

H. Scudder. 1965. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*.

Radu Soricut and Daniel Marcu. 2003. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 228–235.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The penn chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*.

Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. 2021. Dependency-based syntax-aware word representations. *Artificial Intelligence*, 292:103427.

A Brief Introduction of RST

As a prevalent theory in discourse parsing, RST has been studied since early (Mann and Thompson, 1987, 1988) and has been broadly developed (Carlson et al., 2001; Carlson and Marcu, 2001; Soricut and Marcu, 2003). Within the RST framework, a clause is primarily regarded as an EDU, and its boundaries are determined using lexical and syntactic indicators. Each EDU involved in a relationship is attributed with a rhetorical status or nuclearity assignment, which characterizes its semantic role in the overall discourse structure.

The RST framework distinguishes between two types of relations: mononuclear and multinuclear. Mononuclear relations consist of two units, namely the nucleus and the satellite, whereas multinuclear relations involve two or more units of equal importance. A total of 53 mononuclear and 25 multinuclear relations can be used for tagging the RST corpus. These 78 relations can be categorized into 16 distinct classes that share common rhetorical meanings. Furthermore, to establish a clear structure within the tree, three additional relations are utilized, namely textual-organization, span, and same-unit. Our inter-EDU dependency follows most of the coarse-grained relations, and the "topic-comment" relation is refined in four classes to fit the dialogue scenario.

B Dependencies for Dialogue

Different from Carlson et al. (2001), we weaken the concept of nucleus and satellite in our dialogue-level dependencies. For mononuclear relations, an arc is emitted from the nucleus to the satellite. For the multinuclear ones, the arc is always emitted from above (left) to below (right). The landing point of the arc is the core semantic word (always a predicate) of a discourse unit. To satisfy the single root and single head constraints, we keep only the dummy root of the first utterance, and the ones of the other utterances are replaced by inter-utterance links. To adapt the customer service scenario in our dialogue data, we add the "req-proc" label corresponding to a situation, where one party proposes a requirement, and the other proposes to handle it. A model is encouraged to identify the more difficult relationships. Thus, when annotators find that two relationships appear to be appropriate, the more difficult one should be chosen.

Table 4 shows the meaning and quantity of each label. It can be seen that the labels of "root", "obj",

Label	Meaning	Train	Test
root	root	1167	20086
sasubj-obj	same subject, object	14	325
sasubj	same subject	119	2234
dfsubj	different subject	21	373
subj	subject	693	11365
subj-in	inner subject	29	425
obj	object	1264	22752
pred	predicate	84	1508
att	attribute modifier	976	18685
adv	adverbial modifier	1472	25073
cmp	complement modifier	196	3307
coo	coordination	85	1403
pobj	preposition object	198	3540
iobj	indirect-object	15	433
de	de-construction	174	2714
adjct	adjunct	1158	20799
app	appellation	98	1569
exp	explanation	79	1354
punc	punctuation	1238	20966
frag	fragment	10	127
repet	repetition	39	534
attr	attribution	26	336
bckg	background	33	741
cause	cause	27	391
comp	comparison	8	210
cond	condition	27	755
cont	contrast	14	368
elbr	elaboration	696	12199
enbm	enablement	14	286
eval	evaluation	4	139
expl	explanation	18	333
joint	joint	26	480
manner	manner-means	7	105
rstm	restatement	25	473
temp	temporal	41	597
tp-chg	topic-change	6	74
prob-sol	problem-solution	26	441
qst-ans	question-answer	229	3701
stm-rsp	statement-response	381	6617
req-proc	requirement-process	63	1127

Table 4: Statistics of all the dependency relations.

"att", "adv", "adjct", and "punc" are with leading numbers in syntax dependencies. In discourse dependencies, the "elbr" label is the most numerous of the inner-utterance dependencies, and the "stm-rsp" is the most in quantity in inter-utterance ones.

C Implemented Details of MLM

We use the large-scale unlabeled dialogue data $\mathcal{D}$ to train a signal detection model. The model contains an encoder layer of ELECTRA (Clark et al., 2020) and a linear layer as the decoder to project the hidden vector to a vocabulary space. Inspired by prompt-based learning, we set a template as "The word that expresses the signal of discourse dependency is: [mask] [mask] [mask]" in Chinese. The template is as a prefix of input $\boldsymbol{x}$ to obtain the prompt $\boldsymbol{x}'$. As there are many signal words and they are in Chinese, it is difficult to show them and thus we present them in publicly available code. We average the output probabilities in [mask] positions to obtain the signal word distribution. The probability of dropping a word is 0.2, and the one of dropping the signal word is 0.7. We set the epoch number to 2. The other hyper-parameters are the same as section 4.1.

D Impact of Post-Transformation

Table 5 shows the impact of *PostTran*. It can be observed that *PostTran* can benefit the most situations of discourse dependency parsing, both without and with *PreTran*. We think this is because *PostTran* determines some ambiguous predictions by detected signals. Meanwhile, we find that *PostTran* sometimes impairs the performance of inner-EDU parsing. This may be owing to incorrect signals or insufficiently comprehensive rules.

E Detailed Results of Few-shot Settings

Table 6 shows the details of results in few-shot settings. The definitions of the symbols are consistent with those above. It can be seen that the tends of UAS and LAS are extremely in line.

Training Data	Method	Inner		Inter	
		UAS	LAS	UAS	LAS
$\mathcal{S}$	-	87.19	83.77	/	/
	+*PostTran*	87.37	83.14	**65.73**	**49.94**
$\mathcal{S}_\tau$	-	**87.38**	**83.95**	65.57	49.78
	+*PostTran*	87.32	83.16	65.48	49.71
$f_s(\mathcal{D})$	-	86.13	82.90	64.21	48.50
	+*PostTran*	86.18	82.13	64.31	48.76
$f_t(\mathcal{D})$	-	86.11	82.83	63.84	48.23
	+*PostTran*	86.11	82.14	63.93	48.38
$(f_s + f_t)(\mathcal{D})$	-	86.48	82.46	64.58	48.92
	+*PostTran*	86.48	82.46	64.75	49.03
$\mathcal{S} + f_s(\mathcal{D})$	-	86.20	82.99	/	/
	+*PostTran*	86.48	82.48	64.76	48.84
$\mathcal{S}_\tau + f_t(\mathcal{D})$	-	86.73	82.85	64.88	48.62
	+*PostTran*	86.72	82.68	64.93	49.02
$\mathcal{S} + \eta(f_s(\mathcal{D}))$	-	87.97	84.81	/	/
	+*PostTran*	88.05	84.14	66.29	50.47
$\mathcal{S}_\tau + \eta(f_t(\mathcal{D}))$	-	88.13	84.92	65.83	50.20
	+*PostTran*	88.08	84.13	65.88	50.27
$\mathcal{S} + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$	-	88.02	84.78	65.84	50.18
	+*PostTran*	88.12	84.24	66.34	50.62
$\mathcal{S}_\tau + \eta(f_s(\mathcal{D}) + f_t(\mathcal{D}))$	-	88.09	**84.98**	66.45	50.60
	+*PostTran*	**88.22**	84.34	**66.48**	**50.78**

Table 5: The impact of *PostTran*.

Training Data	Inner		Inter	
	UAS	LAS	UAS	LAS
5-shot	44.51±4.48	40.77±4.84	35.41±3.76	25.61±2.92
$+\mathcal{S}$	88.72	85.01	65.80	50.02
$+\mathcal{S}_\tau$	88.76	85.05	65.88	50.09
$+\mathcal{S} + \eta\left(f_s\left(\mathcal{D}\right)\right)$	89.28	85.43	66.45	50.58
$+\mathcal{S}_\tau + \eta\left(f_t\left(\mathcal{D}\right)\right)$	88.98	85.22	66.56	50.78
$+\mathcal{S}_\tau + \eta\left(f_s\left(\mathcal{D}\right) + f_t\left(\mathcal{D}\right)\right)$	**89.15**	**85.55**	**67.57**	**51.12**
10-shot	59.01±4.86	55.83±4.94	45.30±5.72	30.98±3.86
$+\mathcal{S}$	89.34	85.83	66.94	50.98
$+\mathcal{S}_\tau$	89.27	85.69	67.09	51.13
$+\mathcal{S} + \eta\left(f_s\left(\mathcal{D}\right)\right)$	89.35	85.70	67.52	51.56
$+\mathcal{S}_\tau + \eta\left(f_t\left(\mathcal{D}\right)\right)$	89.45	85.81	67.47	51.61
$+\mathcal{S}_\tau + \eta\left(f_s\left(\mathcal{D}\right) + f_t\left(\mathcal{D}\right)\right)$	**89.50**	**85.88**	**68.08**	**51.88**
20-shot	73.51±1.31	70.44±1.28	55.51±2.70	42.30±2.13
$+\mathcal{S}$	90.22	86.80	68.30	52.04
$+\mathcal{S}_\tau$	90.19	86.77	68.21	52.01
$+\mathcal{S} + \eta\left(f_s\left(\mathcal{D}\right)\right)$	**90.74**	**87.03**	69.10	52.51
$+\mathcal{S}_\tau + \eta\left(f_t\left(\mathcal{D}\right)\right)$	90.16	86.75	68.97	52.69
$+\mathcal{S}_\tau + \eta\left(f_s\left(\mathcal{D}\right) + f_t\left(\mathcal{D}\right)\right)$	90.56	86.92	**69.20**	**52.86**
50-shot	85.36±0.22	82.64±0.25	66.39±0.98	51.00±0.66
$+\mathcal{S}$	91.62	88.20	70.93	54.42
$+\mathcal{S}_\tau$	**91.75**	**88.28**	71.00	54.53
$+\mathcal{S} + \eta\left(f_s\left(\mathcal{D}\right)\right)$	91.70	88.18	72.05	55.50
$+\mathcal{S}_\tau + \eta\left(f_t\left(\mathcal{D}\right)\right)$	91.68	88.12	72.35	55.61
$+\mathcal{S}_\tau + \eta\left(f_s\left(\mathcal{D}\right) + f_t\left(\mathcal{D}\right)\right)$	91.74	88.20	**72.53**	**55.73**

Table 6: The details of few-shot results.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations section, below section 7 (conclusion)

☑ A2. Did you discuss any potential risks of your work?
Ethics Statement section

☑ A3. Do the abstract and introduction summarize the paper's main claims?
abstract and introduction

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 2

☑ B1. Did you cite the creators of artifacts you used?
Section 2

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 2 and Ethics Statement

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Ethics Statement

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 2 and Ethics Statement

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 2 and Appendix A

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 2

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 4, Appendix B, and Appendix D

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 2

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Section 2 and appendix A

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Section 2 and Ethics Statement

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Section 2 and Ethics Statement

☑ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Section 2 and Ethics Statement

☑ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Section 2