Hierarchical Macro Discourse Parsing Based on Topic Segmentation

Feng Jiang, Yaxin Fan, Xiaomin Chu, Peifeng Li, Qiaoming Zhu*, Fang Kong
School of Computer Science and Technology, Soochow University, China
{fjiang, yxfansupery}@stu.suda.edu.cn
{xmchu, pfli, qmzhu, kongfang}@suda.edu.cn

Abstract

Hierarchically constructing micro (i.e., intra-sentence or inter-sentence) discourse structure trees using explicit boundaries (e.g., sentence and paragraph boundaries) has been proved to be an effective strategy. However, it is difficult to apply this strategy to document-level macro (i.e., inter-paragraph) discourse parsing, the more challenging task, due to the lack of explicit boundaries at the higher level. To alleviate this issue, we introduce a topic segmentation mechanism to detect implicit topic boundaries and then help the document-level macro discourse parser to construct better discourse trees hierarchically. In particular, our parser first splits a document into several sections using the topic boundaries that the topic segmentation detects. Then it builds a smaller and more accurate discourse sub-tree in each section and sequentially forms a whole tree for a document. The experimental results on both Chinese MCDTB and English RST-DT show that our proposed method outperforms the state-of-the-art baselines significantly.

Introduction

In a coherent document, discourse units (e.g., clauses, sentences, or paragraphs) are tightly connected semantically. Discourse parsing seeks to identify the nuclearity and relationship between discourse units and discover the inner structure of a whole document the units form. It has been shown to be beneficial to many NLP applications, such as question answering (Sadek and Meziane 2016), summarization (Cohan and Goharian 2018) and reading comprehension (Mihaylov and Frank 2019).

As one of the most influential theories in discourse parsing, Rhetorical Structure Theory (RST) (Mann and Thompson 1987) represents a document as a hierarchical structure known as a Discourse Tree (DT). In the literature, various RST-style corpora have been built, such as RST Discourse Treebank (RST-DT) (Carlson, Marcu, and Okurowski 2003) and Macro Chinese Discourse TreeBank (MCDTB) (Jiang et al. 2018b). Commonly, RST-style (document-level) discourse parsing can be divided into micro and macro level. The former studies the intra- or inter-sentence relationship, while the latter studies the discourse relationship among paragraphs or chapters (Van Dijk 1976), which focuses on understanding the full text from the higher-level perspective. Figure 1 is an example of the macro discourse tree. Macro discourse parsing plays an important role in document-level discourse parsing. It reveals the topic and the overall structure of an document from a higher level and is conducive to clarify its organization. Kobayashi et al. (2020) pointed out that the accurate macro discourse parsing is important to obtain a good discourse dependency tree for improving the performance of the downstream NLP tasks.

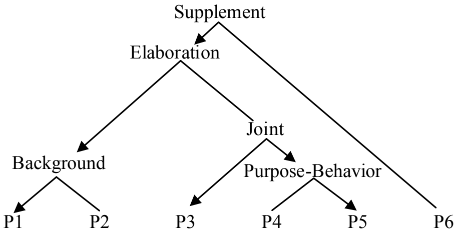


Figure 1: An example of the macro discourse tree. The six leaves (P1-P6) are paragraphs, which are called paragraph-level elementary discourse units (PDUs) in this paper. The directed edge indicates that the node is a nucleus, and the undirected edge indicates that the node is a satellite in nuclearity. The internal node has the relation label that is the relationship between two discourse units.

However, compared with micro-level success (Lin et al. 2019; Liu et al. 2019; Zhang et al. 2020), the macro discourse parsing (Sporleder and Lascarides 2004; Jiang et al. 2018a) faces more challenges because of the larger size and number of discourse units and fewer connectives between them. For example, the average token length of leaf nodes in Chinese MCDTB and English RST-DT is 22 and 8 at the micro level, respectively, while at the macro level, these figures increase to 100 and 52. Even one of the state-of-the-art models (Wang, Li, and Wang 2017) still suffers from significant performance degradation at macro level (micro-level vs. macro-level: 85.11% vs. 37.40% [1]). This result shows that it is essential to improve macro discourse parsing to

*Corresponding author

[1] We used the published source code (https://github.com/yizhongw/StageDP) to reproduce the experiments and evaluated the results by the method of Morey, Muller, and Asher (2017).

build a better complete discourse parser. In this paper, we mainly focus on Chinese macro discourse parsing which is a more challenging task because it has fewer connectives (Chinese MCDTB vs. English RST-DT: 2.99% vs. 13.51%) and longer discourse units.

Constructing discourse structure trees hierarchically (Joty et al. 2013; Feng and Hirst 2014; Kobayashi et al. 2019, 2020) has been proved to be an effective strategy that can better deal with more complex discourse structures. It divides discourse parsing into different granularity levels through explicit boundaries (e.g., sentence boundaries, and paragraph boundaries), and then builds discourse sub-trees within each level to recursively form a complete discourse structure tree. However, it is difficult to apply this method to macro discourse parsing due to the lack of explicit inter-paragraph boundaries at this higher level.

To alleviate this issue, we propose a hierarchical Macro Discourse Parser based on Topic Segmentation (MDParser-TS). Unlike existing methods using those explicit sentence or paragraph boundaries that exist in texts, we use the topic boundary, an implicit inter-paragraph boundary that our topic segmentation model detects, to constrain the construction of document-level macro discourse structure trees. In particular, MDParser-TS first detects the topic boundaries in a document and splits the document into several sections by the topic boundaries. Then it builds a smaller and more accurate discourse sub-tree in each section where the discourse units have the same topic, and sequentially forms a whole tree for a document. Experiments conducted on both Chinese MCDTB and English RST-DT show that our MDParser-TS using the topic segmentation outperforms the state-of-the-art baselines significantly.

Related Work

Discourse Parsing

English RST-DT (Carlson, Marcu, and Okurowski 2003) is one of the popular discourse corpora (Subba and Di Eugenio 2009; Zeldes 2017) that annotates the discourse structure, nuclearity, and relationship for the whole document. Hernault et al. (2010) proposed HILDA that first built a complete discourse structure tree without any boundaries. Recently, constructing micro discourse structure trees using explicit boundaries has been proved to be an effective strategy. Feng and Hirst (2014) and Joty, Carenini, and Ng (2015) used sentence boundaries to model the parser at intra- and inter-sentence separately to build a better discourse tree hierarchically. Kobayashi et al. (2019, 2020) further utilize paragraph boundaries to building discourse trees and proved that the finer the division, the better the performance.

Recently, Liu et al. (2019) and Lin et al. (2019) introduced pointer networks to micro (intra-sentence) discourse parser that got close to human performance. However, only a few work focused on macro discourse parsing. After pruning and revising the original discourse trees in RST-DT, Sporleder and Lascarides (2004) built macro discourse trees on the bottom-up algorithm.

In Chinese, a few discourse corpora (Li et al. 2014; Zhou and Xue 2015) were annotated at micro level. To the best of our knowledge, MCDTB (Jiang et al. 2018b) is the only available macro Chinese discourse corpus. The bottom-up algorithm (Chu et al. 2018; Jiang et al. 2018a) was the earliest applied to construct macro discourse structure trees. Recently, Zhou et al. (2019) built discourse trees with multi-view and word-pair similarity via the shift-reduce algorithm. None of the above work constructed the macro discourse structure tree hierarchically.

Topic Segmentation

Topic Segmentation (TS) aims to unveil the inherent content structure in a multi-paragraph document and identify the potential sections (i.e., sequential paragraphs with coherent topics) of the document (Zhang et al. 2019). Both supervised and unsupervised methods have been proposed for this task. One kind of unsupervised method is based on lexical cohesion (Hearst 1997; Galley et al. 2003), and the other is based on topic models such as LDA (Riedl and Biemann 2012). Supervised methods mainly use neural network models such as semantic matching networks (Alemi and Ginsparg 2015; Wang et al. 2017), pointer networks (Li, Sun, and Joty 2018), transformer networks (Glavaš and Somasundaran 2020) and S-LSTM (Barrow et al. 2020).

Hierarchical Discourse Parsing Based on Topic Segmentation

As pointed out in the introduction, one crucial issue in macro discourse parsing is that it is difficult to build a macro discourse tree hierarchically due to the lack of explicit inter-paragraph boundaries. To alleviate this issue, we integrate topic segmentation into macro discourse parsing and obtain topic boundaries as implicit boundaries through topic segmentation. Then the parser can build macro structure trees hierarchically using these implicit boundaries.

Thus, we propose a hierarchical Macro Discourse Parser based on Topic Segmentation (MDParser-TS), and its high-level illustration is shown in Figure 2. Our MDParser-TS mainly includes three stages: (1) topic segmentation for a document; (2) discourse parsing within sections; (3) discourse parsing between sections. In the stage of topic segmentation (stage 1), MDParser-TS first segment a document into several sections according to the topic consistency. In the stage of discourse parsing (stage 2 and 3), MDParser-TS first constructs the structure naked tree inside and between sections and then recognizes the nuclearity and relationship of the node in the tree individually.

Motivation

Previous studies (Feng and Hirst 2014; Joty, Carenini, and Ng 2015; Kobayashi et al. 2019, 2020) have shown that it is an effective strategy to build discourse structure trees hierarchically. They use explicit boundaries (e.g., sentence and paragraph boundaries) that exist in the text to divide the process of discourse parsing into different levels: building discourse trees in a sentence, between sentences, and between paragraphs separately. Only smaller discourse subtrees need to be built at each level, thereby reducing the difficulty of constructing a more complex discourse structure

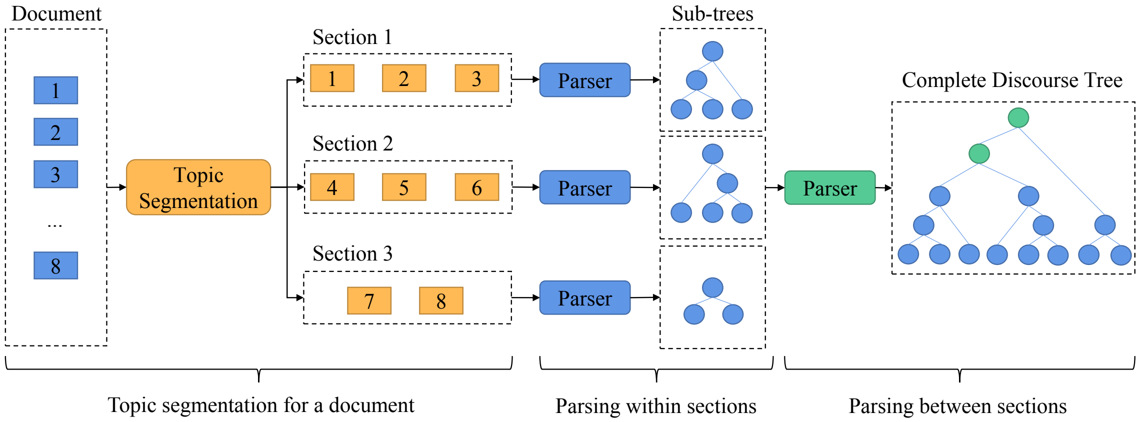


Figure 2: The framework of macro discourse parser based on topic segmentation (MDParser-TS), using an example of parsing a document that contains eight PDUs (1-8).

tree directly. However, there are no explicit inter-paragraph boundaries that can be utilized at the macro level, while the sentence and paragraph boundaries exist in texts that can be used at the micro level.

Therefore, we seek for such inter-paragraph boundaries, which conduce to perform macro discourse parsing better. According to the macrostructure theory (Van Dijk 1976), a document has a macro-structure at a higher level (above the paragraphs), which reflects the overall thought and thread of the document. When a sequence of discourse units support and surround a central topic, they form a macro structure. Besides, Stede (2011) points out that in addition to the genre-based structure in the text, there is a topic-based structure that is also helpful for discourse parsing. It reveals that long stretches of running text can sensibly be broken into smaller segments whose "hanging together" is motivated by their dealing with a common topic.

Inspired by the above theory, we introduce a topic segmentation mechanism to detect implicit topic boundaries. As shown in Figure 3, the document contains 11 paragraphs, which can be split into four sections according to their topics, each of which is relatively cohesive and independent. It is difficult to construct such a complex discourse structure tree directly due to longer elementary discourse units (i.e., PDUs at the macro level) and more discourse units in the tree. However, it is easier to build the discourse sub-tree in each section and sequentially build the whole discourse tree between sections, if the document is split into four sections (S1-S4) with the boundaries PDU_1, PDU_4, PDU_7 and PDU_{11} according to the topics.

Preparing Data for Topic Segmentation

Therefore, we use topic segmentation to detect such topic boundaries: there are one or more continuous discourse units within each topic section, and a discourse sub-tree corresponds to a topic. Due to the poor performance of unsupervised methods (Galley et al. 2003; Riedl and Biemann 2012), we introduce a supervised model to segment a document into sections by their topics, thereby achieving better performance with labeled data to help the downstream task.

To obtain the labeled data required for training the supervised model in topic segmentation, we propose a data conversion method on discourse structure tree to obtain the annotated topic boundaries as follows: (1) a topic corresponds to a sub-tree; (2) the number of PDUs in a sub-tree does not exceed half of that in the whole discourse tree of a document. These two simple rules ensure that the size of the generated topic is moderate and can be coordinated with its original discourse structure tree. To verify that this method satisfies these two constraints, we manually checked the conversion results on the training set of the MCDTB corpus and find out that almost 96% of the sub-trees are consistent with topics, because the topic information is also helpful to construct the upper level of discourse trees.

Following the above conversion method, we convert the annotated documents in the training set of discourse parsing to those with topic boundaries and use them as the training set of the topic segmentation task. As exemplified in Figure 3, the document comprising 11 PDUs will be converted to four sections (S1-S4) that have specific topics separately and all of the discourse sub-trees contain no more than 5 PDUs.

Model Specifics for Topic Segmentation

Following previous work (Li, Sun, and Joty 2018), we treat topic segmentation as a sequence labeling task. Formally, given an input paragraph sequence $P = (p_1, p_2, ..., p_N)$ of length N where p_i is the ith paragraph in the document, our ultimate goal is to split the input sequence into contiguous segments by identifying the topic boundaries.

Considering the relationship between two adjacent paragraphs not only depends on themselves, but also derives from other paragraphs around them, we propose a Triple semantic Matching model based on BERT (TM-BERT), as shown in Figure 4, as a local model of the sliding window mechanism to predict whether a paragraph is a topic boundary. Different from the vanilla BERT on semantic matching two adjacent paragraphs, TM-BERT using three consecutive paragraphs as the input, not only matches the semantics of

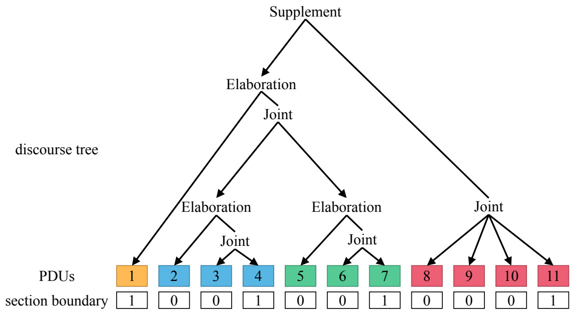


Figure 3: An example of topic segmentation (chtb 0234 in MCDTB). This document can be split into four sections (S1-S4) according to four topics as follows. S1 (PDU_1): the Congress adopted the Arbitration Law and the Audit Law; S2 (PDU_2-PDU_4): the purpose and content of Arbitration Law; S3 (PDU_5-PDU_7): the purpose and content of the Audit Law; S4 (PDU_8-PDU_{11}): other contents adopted by the Congress.

consecutive paragraph pair but also matches that of across-paragraph pair.

In the encoding layer, we first encode the two adjacent paragraphs by BERT (Devlin et al. 2018). The input of BERT BI consists of three parts: $Token$, $Segment$ and $Position$ of two matching paragraphs. The $Token$ is the vector of all of the index of the words in two adjacent paragraphs; the $Segment$ is the vector of the index of the paragraph that the word belongs to; the $Position$ is the vector of the word order in the sequence that consists of two adjacent paragraphs. BM is the embedding of $[CLS]$ position in the output of BERT. Moreover, we propose a triple semantic matching mechanism that employs the pairs of $\alpha(p_{i-1}, p_i)$ and $\beta(p_i, p_{i+1})$ for semantic matching but also matches the across-paragraph pair $\gamma(p_{i-1}, p_{i+1})$, as shown in Equation 1. Besides, it is worthwhile that we do not add any hand-crafted features into the model to ensure its universality.

$$BM_k = BERT(BI_k),\ k \in \{\alpha, \beta, \gamma\} \tag{1}$$

In the decoding layer, we concentrate the triple semantic match output $(BM_\alpha, BM_\beta, BM_\gamma)$ and feed them to a Softmax layer, as shown in Equation 2. During training, we use the Adam optimizer to optimize the network parameters by maximizing the log-likelihood loss function between the predicted label and the true label.[2]

$$y = Softmax(con(BM_\alpha, BM_\beta, BM_\gamma)) \tag{2}$$

In particular, for the topic segmentation, if there is a title, we could consider it as the overall topic in a document. Therefore, we propose the TM-BERT (Topic) model, a variant of TM-BERT, which replaces p_{i+1} with document title.

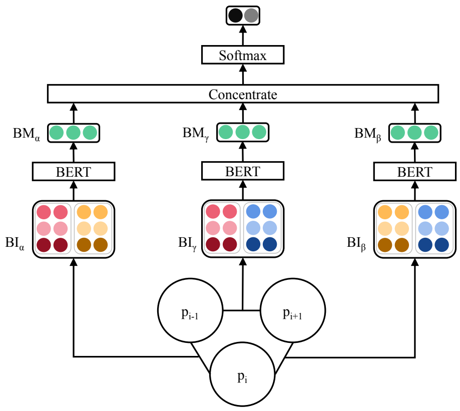


Figure 4: The model architecture of TM-BERT.

Model Specifics for Discourse Parsing

We use the popular transition-based method (shift-reduce algorithm) to build a discourse structure tree. The parsing process is modeled as a sequence of shift and reduce actions, as shown in Figure 5. We also use TM-BERT of topic segmentation as a local model in both discourse parsing within and between sections because it naturally fits with the shift-reduce algorithm that relies on three DUs (S1, S2, and Q1), as same as the input of topic segmentation.

Specifically, at each step, the parser generates or does not generate a new span, according to the prediction obtained from the structure classifier TM-BERT$_s$ (PD_s is the span probability distribution), which uses three DUs S1, S2, and Q1 as the input. The parser then applies a nuclearity classifier TM-BERT$_n$ and a relation classifier TM-BERT$_r$ to predict nuclearity labels (PD_n is the nuclear probability distribution) and relation labels (PD_r is relation probability distribution) for the new span separately as follows:

$$PD_t = TM\text{-}BERT_t(BI_\alpha, BI_\beta, BI_\gamma), t \in s, n, r \tag{3}$$

where TM-BERT$_t$ uses the same parameters and optimizer as TM-BERT in topic segmentation.

To prevent impossible collocations between nuclearity and relations (e.g., it is impossible to match Nucleus-Nucleus with Elaboration in reality) and save GPU computing resources, we use a nuclearity-relation co-occurrence matrix instead of joint learning to optimize the prediction of TM-BERT$_n$ and TM-BERT$_r$. We first multiply PD_n with PD_r to obtain the nuclearity-relation probability matrix (PM_{nr}) as follows:

$$PM_{nr} = PD_r \times PD_n^T \tag{4}$$

We use the nuclearity-relation co-occurrence matrix (CM_{nr}) (each element is 0/1) to constrain PM_{nr} to $PM_{nr}^{'}$ with mask operations to set the probability of impossible labels as zero, as shown in Equation 5. This matrix is based on the co-occurrence of nuclearity and relation in the training set. In a discourse tree, each DU (non-leaf node) has two labels, nuclearity and relation. We enumerate all the

[2] We use Keras library to implement our model and use keras_bert package (https://github.com/CyberZHG/keras-bert) to load BERT parameters. The key hyper parameters are following: batch-size=2, epoch=5, max-length=512, and learning-rate=1e-5.

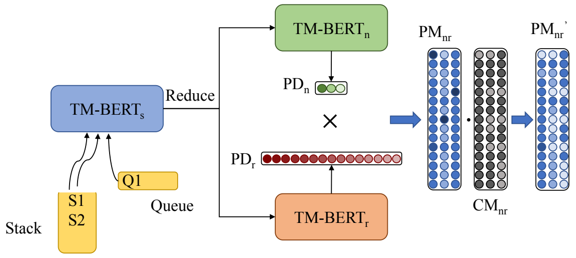


Figure 5: The process of our macro discourse parser, where TM-BERT$_s$, TM-BERT$_n$, and TM-BERT$_r$ are the TM-BERT models for discourse tree construction, nuclearity recognition, and relation classification, respectively.

co-occurrence of nuclearity and relation for each DU in the training set to construct the matrix CM, which is a 0/1 matrix. $CM[n][r]$=1 if the nuclearity n and relation r co-occurs in the training set; $CM[n][r]$=0 if the nuclearity n and Relation r do not co-occur in the training set. For example, if a DU has the relation label Elaboration and the nuclearity label Nucleus-Satellite (NS), $CM[NS][Elaboration]$ is set to 1. Then we select the highest probability of Nuclearity (P_n) and Relation (P_r) from $PM_{nr}^{'}$.

$$PM_{nr}^{'} = PM_{nr} \cdot CM_{nr} \tag{5}$$

Experimentation

Data and Metrics

In discourse parsing, our experiments are evaluated on the Macro Chinese Discourse Treebank (MCDTB) (Jiang et al. 2018b) that contains 720 news annotated document-level macro discourse trees. Following previous work (Zhou et al. 2019), we transform the non-binary tree of the original data into the right-binary tree. There are 80% data for the training set and 20% data for the test set. In particular, to balance the training set and the test set, we divide the documents containing different numbers of paragraphs into the training set (576 documents) and the test set (144 documents) according to the proportion. Finally, there are 3194 paragraphs in the training set and 791 paragraphs in the test set, and we randomly select 10% of the training set as the validation set.

We use the fairer method (Morey, Muller, and Asher 2017) to evaluate the experimental results. Following previous work, we report the micro-averaged F_1 score for predicting span attachments in discourse tree construction (Span), span attachments with nuclearity (Nuclearity), and span attachments with relation labels (Relation). Specifically, we evaluate the nuclearity with three classes (*Nucleus-Satellite*, *Satellite-Nucleus*, and *Nucleus-Nucleus*), and we use 15 finer-grained types for evaluation in relation classification.

Experimental Results

We compare our MDParser-TS with the following benchmarks: 1) **Rule-left** and **Rule-right**: we use rules to produce a left- or right-branching tree by always merging the leftmost or rightmost two discourse units, respectively; 2) **LD-CM** (Jiang et al. 2018a): we modified the first structure prediction model LD-CM to build macro discourse trees using a bottom-up algorithm; 3) **MVM** (Zhou et al. 2019): it is the state-of-the-art model for constructing macro discourse trees via the shift-reduce algorithm. Since the above two systems did not recognize the nuclearity and classify the relation of a span, we did that through STGSN (Jiang, Li, and Zhu 2019); 4) **LS19** (Lin et al. 2019): we also introduce this state-of-the-art micro (intra-sentence) English discourse parser to MCDTB by the top-down algorithm; 5) **BERT**: due to the overwhelming impact of BERT in many NLP applications, BERT is selected as the local model for building a discourse tree with the shift-reduce algorithm.

Model	Span	Nuclearity	Relation
Rule-right	39.88	-	-
Rule-left	52.55	-	-
LD-CM	54.71	48.38	26.28
MVM	56.11	47.76	27.67
LS19	56.25	46.21	28.75
BERT	57.19	48.38	28.44
MDParser-TS	**66.31**	**55.80**	**35.39**

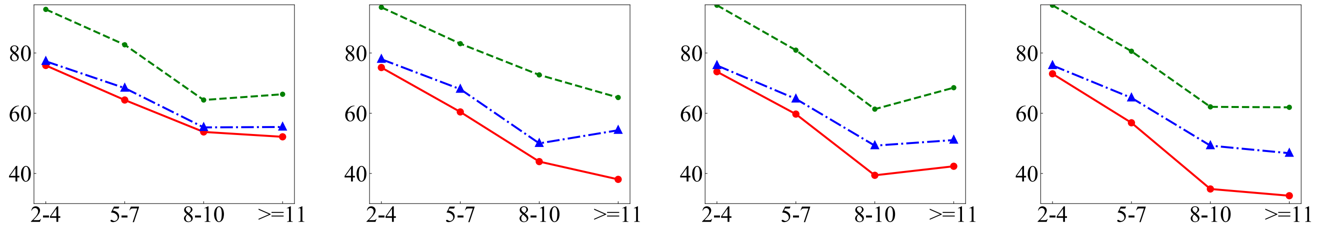
Table 1: The performance comparison between our MDParser-TS and the benchmark systems. We used the t-test with a 95% confidence interval for the significance test and all improvements of MDParser-TS over BERT are significant (p < 0.001). We did three-time experiments and reported the averaged performance.

Table 1 shows the performance comparison between our MDParser-TS and all the baselines. Although LD-CM, MVM and LS19 apply various models to macro discourse parsing, these three models achieved similar performance and their improvements are slight, in comparison with Rule-left. LS19 achieved human performance in micro (intra-sentence) discourse parsing; however, it does not work well at the macro level. This result demonstrates that the macro discourse parsing is still a challenge due to longer elementary discourse units and fewer connectives than those at the micro level. Moreover, BERT's performance is better than the other baselines due to the use of the pre-trained model.

In comparison with the best baseline BERT, our MDParser-TS improves the micro-F_1 score by 9.12, 7.42, and 6.95 on the discourse tree construction (Span), nuclearity recognition (Nuclearity), and relation classification (Relation), respectively. This result verified the effectiveness of our MDParser-TS on macro discourse parsing. Besides, it is worthwhile to emphasize that our topic segmentation mech-

Model	w/o TS	w/ auto TS	w/ golden TS
Rule-right	39.88	53.79 (+13.91)	71.56 (+31.68)
Rule-left	52.55	61.67 (+9.12)	77.59 (+25.04)
LS19	56.25	62.13 (+5.88)	78.52 (+22.27)
BERT	57.19	64.61 (+7.42)	81.14 (+23.95)
MDParser-TS	**63.06**	**66.31 (+3.25)**	**83.77 (+20.71)**

Table 2: The ablation experiments of MDParser-TS and several representative baselines on the Span task.



(a) MDParser-TS (b) BERT (c) LS19 (d) Rule-left

Figure 6: Micro-F_1 scores (Y-axis) on different PDU numbers (X-axis). The red solid line, the blue chain-dotted line and the green dashed line indicate the model w/o TS, w/ auto TS and w/ golden TS, separately.

anism can split a document into several sections by their topics and then reduce the complexity of structure tree construction (Span). Hence, MDParser-TS achieves the highest improvement in discourse tree construction (Span). Since the nuclearity recognition (Nuclearity) and relation classification (Relation) are the downstream tasks of the discourse tree construction (Span), their improvements mainly derive from the success of discourse tree construction.

Analysis

Impact of Topic Segmentation

To explore the impact of topic segmentation on discourse parsing, we conducted ablation experiments to compare the performance of MDParser-TS and the baselines without using topic segmentation (w/o TS), with automatic topic segmentation (w/ auto TS), and with golden topic segmentation (w/ golden TS), as shown in Table 2. It can be observed that automatic topic segmentation brings an absolute improvement of 3.25-13.91 in all various models on the task Span. Employing the automatic topic segmentation, even the simple model Rule-left can achieve 61.67% in micro-F_1, which is better than other baselines without topic segmentation in Table 1. These results ensure that our topic segmentation mechanism has strong applicability and generality. In addition, we get the upper bound of the performance through the golden topic segmentation (using annotated discourse trees to create topic boundaries), as shown in Table 2. It shows that the golden topic segmentation greatly improves all models (20.71-31.68 in micro-F_1), which proves that integrating topic segmentation into discourse parsing is an effective mechanism to improve its performance.

Performance on Different Lengths of Documents

We further analyze the performance of our MDParser-TS in terms of the number of PDUs. Figure 6 shows the micro-F_1 score on Span of MDParser-TS with or without topic segmentation as well as several representative baselines. We can find that various models, including MDParser-TS, have improvements in different lengths with topic segmentation.

Moreover, in those longer documents, topic segmentation can greatly improve the three baselines with lower performance. One reason is that the topic segmentation can constrain the size of the sub-tree that needs to be built in each section. According to the statistics, all documents in the test set are split into at most six sections, and each section contains no more than seven PDUs, which can reduce the cascading errors in the construction of those complex discourse trees (i.e., long documents).

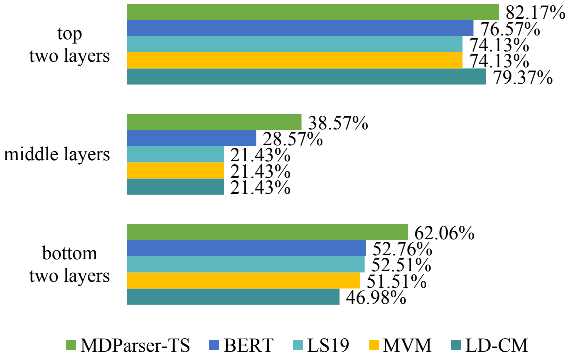


Figure 7: The performance of Span on different layers of discourse trees.

Performance on Different Layers of DT

Figure 7 shows the performance of MDParser-TS and other baselines on different layers of discourse trees, and we can find that MDParser-TS outperforms all baselines with the significant improvements on micro-F_1 score from 2.80 to 8.04 on the top two layers, 10.00-17.14 on the middle layers, and 9.30-15.08 on the bottom two layers, respectively. It shows the performance improvement mainly comes from the bottom two layers and the middle layers.

Compared with constructing macro discourse trees directly, MDParser-TS can construct trees in each section more accurately when the topic segmentation splits the document into different topics (sections). It brings a great improvement on the bottom two layers. Moreover, the implicit topic boundaries can help our MDParser-TS build better middle-level spans by implying the sketch of discourse trees.

In addition, we observed that all models have the best performance on the top two layers, while they have the worst performance on the middle layers. The low performance on the middle layers is due to the complex discourse trees. Those discourse trees with middle layers are more complex than the others because they have more layers. Besides, com-

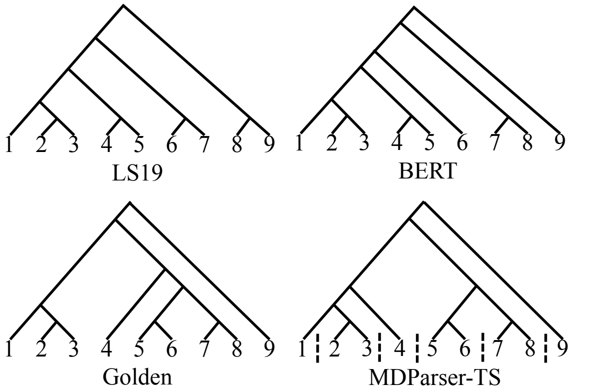


Figure 8: The discourse tree of chtb 0792 parsed by various models. The dashed line indicates the predicted topic boundaries.

pared with the bottom two layers and the top two layers, the relationships between discourse units on the middle layers are more ambiguous.

Case Study

Figure 8 gives an example of chtb 0792 parsed by MDParser-TS and two baselines LS19 and BERT. It shows that LS19 and BERT without topic segmentation are inaccurate when building such a complex discourse tree because they do not consider the overall document structure. Compared with them, the discourse tree built by our MDParser-TS is more similar to the golden tree. With the help of the topic boundaries, MDParser-TS can build the better sub-tree in each section and then obtain the more accurate discourse trees on the bottom layers. Moreover, using those sub-trees in all sections, MDParser-TS builds the whole discourse tree between sections easily due to fewer discourse units.

Error Analysis

The errors of MDParser-TS come from two aspects: the upstream errors in the topic segmentation stage and the errors in the discourse parsing stage. In Table 3, we report the performance of our topic segmentation model TM-BERT with three baselines including LSTM-CRF (Lample et al. 2016), Pointer Networks (Lin et al. 2019; Li, Sun, and Joty 2018) and BERT. We use the following commonly used metrics[3] in topic segmentation to evaluate models: P_k (Beeferman, Berger, and Lafferty 1999), WindowDiff (Pevzner and Hearst 2002), Segmentation Similarity (Fournier and Inkpen 2012) and Boundary Similarity (Fournier 2013). It can be observed that although our model beats other baselines on most indicators, its performance is still low. Therefore, there is still much room for improvement in topic segmentation, which motivates us to do further study on topic segmentation for promoting discourse parsing in future work.

Model	**P_k (%) ↓**	**WD (%) ↓**	**S(%) ↑**	**B(%) ↑**
TM-BERT (Topic)	**21.8**	43.8	**76.2**	**61.8**
TM-BERT	27.2	47.2	75.7	58.5
BERT	38.8	48.0	71.3	44.7
Pointer Networks	27.0	46.1	71.0	53.7
LSTM-CRF	24.9	**40.6**	75.1	54.0

Table 3: Segmentation results on MCDTB. Different from Segmentation Simlarity (S) and Boundary Similarity (B), P_k and WindowDiff (WD) are penalty measures.

Models	**Span**	**Nuclearity**	**Relation**
Rule-left	28.57	-	-
Rule-right	31.17	-	-
SL04	34.29	-	-
WL17	37.40	28.83	18.70
MDParser-TS	40.52	32.99	22.60
MDParser-TS (golden)	63.38	47.53	28.57

Table 4: The performance comparison on the RST-DT at the macro level.

Besides, the errors are also cascaded into discourse parsing. Our MDParser-TS achieves 56.41 and 78.04 in micro-F_1 score within topics and between topics, respectively. Compared with MDParser-TS (golden) using annotated topic boundaries, whose figures are 83.76 and 83.78, respectively, the performance decline of MDParser-TS within the topic is significant (56.41 vs. 83.76) due to the inaccurate topic boundaries.

Performance on English RST-DT

To verify the generalization of the proposed model, we also evaluate our models on the English RST-DT, and the preliminary experimental results are shown in Table 4[4] where **SL04** (Sporleder and Lascarides 2004) is the first model attempted to build discourse trees at the macro level on RST-DT and **WL17** (Wang, Li, and Wang 2017) is one of the state-of-the-art models on RST-DT. In comparison with **WL17**, our MDParser-TS improves the micro-F_1 score by 3.12%, 4.16%, 3.90% on Span, Nuclearity and Relation separately, which proves the effectiveness of our model on English macro discourse parsing. Moreover, similar to the performance of MDParser-TS (golden) on MCDTB, there is a significant improvement of MDParser-TS (golden) on RST-DT. It shows that the topic segmentation is also beneficial to English macro discourse parsing.

Conclusion

In this paper, we introduce a topic segmentation mechanism to detect topic boundaries and then help the discourse parser to better construct discourse structure trees hierarchically. Experimental results on both Chinese MCDTB and English RST-DT show that our MDParsesr-TS outperforms all baselines significantly. Our future work will focus on how to optimize the topic segmentation model and then promote discourse parsing.

[3] https://github.com/cfournie/segmentation.evaluation

[4] To evaluate discourse parsing at the macro level, we prune and revise the original discourse trees in RST-DT to macro level, following Sporleder and Lascarides (2004). The performance is much lower than that of MCDTB because the average number of paragraphs in RST-DT is larger than that in MCDTB (9.99 vs. 5.53).

Acknowledgements

The authors would like to thank the three anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (No. 61836007, 61772354 and 61773276.), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

Alemi, A. A.; and Ginsparg, P. 2015. Text segmentation based on semantic word embeddings. *arXiv preprint arXiv:1503.05543* .

Barrow, J.; Jain, R.; Morariu, V.; Manjunatha, V.; Oard, D.; and Resnik, P. 2020. A Joint Model for Document Segmentation and Segment Labeling. In *ACL*, 313–322. Online: Association for Computational Linguistics.

Beeferman, D.; Berger, A.; and Lafferty, J. 1999. Statistical Models for Text Segmentation. *Machine Learning* 34(1-3): 177–210.

Carlson, L.; Marcu, D.; and Okurowski, M. E. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory 85–112.

Chu, X.; Jiang, F.; Zhou, Y.; Zhou, G.; and Zhu, Q. 2018. Joint modeling of structure identification and nuclearity recognition in macro Chinese discourse treebank. In *COLING*, 536–546. Association for Computational Linguistics.

Cohan, A.; and Goharian, N. 2018. Scientific document summarization via citation contextualization and scientific discourse. *International Journal on Digital Libraries* 19(2-3): 287–303.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* .

Feng, V. W.; and Hirst, G. 2014. A linear-time bottom-up discourse parser with constraints and post-editing. In *ACL*, volume 1, 511–521. Association for Computational Linguistics.

Fournier, C. 2013. Evaluating Text Segmentation using Boundary Edit Distance. In *ACL*, volume 1, 1702–1712. Sofia, Bulgaria: Association for Computational Linguistics.

Fournier, C.; and Inkpen, D. 2012. Segmentation Similarity and Agreement. In *NAACL*, 152–161. Montréal, Canada: Association for Computational Linguistics.

Galley, M.; McKeown, K. R.; Fosler-Lussier, E.; and Jing, H. 2003. Discourse Segmentation of Multi-Party Conversation. In *ACL*, 562–569. Sapporo, Japan: Association for Computational Linguistics.

Glavaš, G.; and Somasundaran, S. 2020. Two-Level Transformer and Auxiliary Coherence Modeling for Improved Text Segmentation. In *AAAI*. Association for the Advancement of Artificial Intelligence.

Hearst, M. A. 1997. Text Tiling: Segmenting Text into Multi-paragraph Subtopic Passages. *Computational Linguistics* 23(1): 33–64.

Hernault, H.; Prendinger, H.; Ishizuka, M.; et al. 2010. HILDA: A discourse parser using support vector machine classification. *Dialogue & Discourse* 1(3): 1–33.

Jiang, F.; Li, P.; Chu, X.; Zhu, Q.; and Zhou, G. 2018a. Recognizing macro Chinese discourse structure on label degeneracy combination model. In *CCF International Conference on Natural Language Processing and Chinese Computing*, 92–104. Springer.

Jiang, F.; Li, P.; and Zhu, Q. 2019. Joint Modeling of Recognizing Macro Chinese Discourse Nuclearity and Relation Based on Structure and Topic Gated Semantic Network. In *CCF International Conference on Natural Language Processing and Chinese Computing*, 276–286. Springer.

Jiang, F.; Xu, S.; Chu, X.; Li, P.; Zhu, Q.; and Zhou, G. 2018b. MCDTB: A Macro-level Chinese Discourse TreeBank. In *COLING*, 3493–3504. Association for Computational Linguistics.

Joty, S.; Carenini, G.; Ng, R.; and Mehdad, Y. 2013. Combining intra- and multi-sentential rhetorical parsing for document-level discourse analysis. In *ACL*, volume 1, 486–496. Association for Computational Linguistics.

Joty, S.; Carenini, G.; and Ng, R. T. 2015. Codra: A novel discriminative framework for rhetorical analysis. *Computational Linguistics* 41(3): 385–435.

Kobayashi, N.; Hirao, T.; Kamigaito, H.; Okumura, M.; and Nagata, M. 2020. Top-down RST Parsing Utilizing Granularity Levels in Documents. In *AAAI*. Association for the Advancement of Artificial Intelligence.

Kobayashi, N.; Hirao, T.; Nakamura, K.; Kamigaito, H.; Okumura, M.; and Nagata, M. 2019. Split or Merge: Which is Better for Unsupervised RST Parsing? In *EMNLP-IJCNLP*, 5801–5806. Association for Computational Linguistics.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural Architectures for Named Entity Recognition. In *NAACL*, 260–270. San Diego, California: Association for Computational Linguistics.

Li, J.; Sun, A.; and Joty, S. 2018. SegBot: A Generic Neural Text Segmentation Model with Pointer Network. In *IJCAI*, 4166–4172. International Joint Conferences on Artificial Intelligence Organization.

Li, Y.; Kong, F.; Zhou, G.; et al. 2014. Building Chinese discourse corpus with connective-driven dependency tree structure. In *EMNLP*, 2105–2114. Association for Computational Linguistics.

Lin, X.; Joty, S.; Jwalapuram, P.; and Bari, M. S. 2019. A unified linear-time framework for sentence-Level discourse parsing. In *ACL*, 4190–4200. Association for Computational Linguistics.

Liu, L.; Lin, X.; Joty, S.; Han, S.; and Bing, L. 2019. Hierarchical Pointer Net Parsing. In *EMNLP-IJCNLP*, 1006–1016. Association for Computational Linguistics.

Mann, W. C.; and Thompson, S. A. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute.

Mihaylov, T.; and Frank, A. 2019. Discourse-Aware Semantic Self-Attention for Narrative Reading Comprehension. In *EMNLP-IJCNLP*, 2541–2552. Hong Kong, China: Association for Computational Linguistics.

Morey, M.; Muller, P.; and Asher, N. 2017. How much progress have we made on RST discourse parsing? A replication study of recent results on the RST-DT. In *EMNLP*, 1325–1330. Association for Computational Linguistics.

Pevzner, L.; and Hearst, M. A. 2002. A Critique and Improvement of an Evaluation Metric for Text Segmentation. *Computational Linguistics* 28(1): 19–36.

Riedl, M.; and Biemann, C. 2012. TopicTiling: A Text Segmentation Algorithm based on LDA. In *Proceedings of ACL 2012 Student Research Workshop*, 37–42. Jeju Island, Korea: Association for Computational Linguistics.

Sadek, J.; and Meziane, F. 2016. A discourse-based approach for Arabic question answering. *TALLIP* 16(2): 11.

Sporleder, C.; and Lascarides, A. 2004. Combining hierarchical clustering and machine learning to predict high-level discourse structure. In *COLING*, 43–49. Association for Computational Linguistics.

Stede, M. 2011. Discourse processing. *Synthesis Lectures on Human Language Technologies* 4(3): 1–165.

Subba, R.; and Di Eugenio, B. 2009. An effective discourse parser that uses rich linguistic information. In *NAACL-HLT*, 566–574. Association for Computational Linguistics.

Van Dijk, T. A. 1976. Narrative macro-structures. *PTL: A journal for descriptive poetics and theory of literature* 1: 547–568.

Wang, L.; Li, S.; Lv, Y.; and Wang, H. 2017. Learning to Rank Semantic Coherence for Topic Segmentation. In *EMNLP*, 1340–1344. Copenhagen, Denmark: Association for Computational Linguistics.

Wang, Y.; Li, S.; and Wang, H. 2017. A two-stage parsing method for text-level discourse analysis. In *ACL*, volume 2, 184–188.

Zeldes, A. 2017. The GUM corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation* 51(3): 581–612.

Zhang, L.; Xing, Y.; Kong, F.; Li, P.; and Zhou, G. 2020. A Top-down Neural Architecture towards Text-level Parsing of Discourse Rhetorical Structure. In *ACL*, 6386–6395. Online: Association for Computational Linguistics.

Zhang, R.; Guo, J.; Fan, Y.; Lan, Y.; and Cheng, X. 2019. Outline Generation: Understanding the Inherent Content Structure of Documents. In *SIGIR*. Paris, France: Association for Computing Machinery.

Zhou, Y.; Chu, X.; Li, P.; and Zhu, Q. 2019. Constructing Chinese Macro Discourse Tree via Multiple Views and Word Pair Similarity. In *CCF International Conference on Natural Language Processing and Chinese Computing*, 773–786. Springer.

Zhou, Y.; and Xue, N. 2015. The Chinese Discourse TreeBank: a Chinese corpus annotated with discourse relations. *Language Resources and Evaluation* 49(2): 397–431.