Employing Text Matching Network to Recognise Nuclearity in Chinese Discourse

Sheng Xu[1], Peifeng Li[1], Guodong Zhou[1], and Qiaoming Zhu[12]
[1]School of Computer Science and Technology, Soochow University, China
[2]Institute of Artificial Intelligence, Soochow University, China
sxu@stu.suda.edu.cn;{pfli, gdzhou, qmzhu}@suda.edu.cn

Abstract

The task of nuclearity recognition in Chinese discourse remains challenging due to the demand for more deep semantic information. In this paper, we propose a novel text matching network (TMN) that encodes the discourse units and the paragraphs by combining Bi-LSTM and CNN to capture both global dependency information and local n-gram information. Moreover, it introduces three components of text matching, the Cosine, Bilinear and Single Layer Network, to incorporate various similarities and interactions among the discourse units. Experimental results on the Chinese Discourse TreeBank show that our proposed TMN model significantly outperforms various strong baselines in both micro-F1 and macro-F1.

1 Introduction

During the past few years, the focus of Natural Language Understanding (NLU) has shifted from the word/sentence level to the discourse level. A challenging task in NLU is discourse parsing, which involves analysing the relations between discourse units and building the document structure. As one of the most influential discourse theories, Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) defines a document as a collection of Elementary Discourse Units (EDUs) with semantic connections and combines adjacent EDUs with rhetorical relations in a hierarchical way to represent an entire document as a discourse tree.

As a critical subtask in discourse parsing, nuclearity recognition involves identifying the nuclearity between the discourse units and thus being able to extract the main information in a document. According to RST, a discourse relation can be divided into mononuclear and multinuclear. A mononuclear relation holds a nucleus and a satellite, where the nucleus expresses the main textual information and the satellite offers additional information about the nucleus (Stede, 2008), while a multinuclear relation holds two or more discourse units, which are all nuclei. Therefore, three types of nuclearity exist: Nucleus-Satellite if the left subtree is the nucleus and the right subtree is the satellite, Satellite-Nucleus if the order of the satellite and nucleus is inverted, and Nucleus-Nucleus for multinuclear relations.

Nuclearity recognition is helpful in detecting discourse relations (Iruskieta et al., 2014) and extracting the main content of a document, and it is widely used in various NLP tasks, including automatic summarisation (Louis et al., 2010; Marcu, 2000), question answering (Verberne et al., 2007) and information extraction (Zou et al., 2014). Consider the following document as an example:

Example 1: 中国机电产品进出口贸易继续增加 $_{a}$，占总进出口的比重继续上升 $_{b}$。其中，出口五十七点九亿美元 $_{c}$，占总出口的百分之三十二点五 $_{d}$；进口八十五点二亿美元 $_{e}$，占总进口的百分之四十六点四 $_{f}$，均比去年同期有所上升 $_{g}$。 *The import and export trade of China's mechanical and electronic products continues to increase* $_{a}$, *and its proportion of the total imports and exports also continues to rise* $_{b}$. *Among them, the exports amounted to 5.79 billion dollars* $_{c}$, *accounting for 32.5 percent of the total exports* $_{d}$; *and the imports of 8.52 billion dollars* $_{e}$, *accounting for 46.4 percent of the total imports* $_{f}$; *all of them were higher than those in the same period last year* $_{g}$.

Example 1 shows a paragraph that includes seven EDUs (*a*-*g*), and its corresponding nuclearity discourse tree is illustrated in Figure 1, where the leaf nodes (*a*-*g*) in Figure 1 are EDUs and the internal

Proceedings of the 27th International Conference on Computational Linguistics, pages 525–535
Santa Fe, New Mexico, USA, August 20-26, 2018.

nodes refer to relational nodes, which represent the combination of the relevant children. When connecting the parent and child nodes, the directed edge indicates that the child is a nucleus in the relationship and the undirected edge indicates that the child is a satellite.

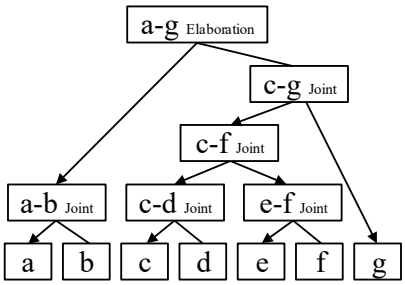

Figure 1: Discourse tree of Example 1.

Starting from the root node, i.e., *a-g* in Figure 1, we can continually select all of the branches labelled as nucleus until the leaf node, i.e., *a* in Figure 1 (中国机电产品进出口贸易继续增加 *The import and export trade of China's mechanical and electronic products continues to increase*), which can be used to represent a summary of this paragraph.

Although there are many studies on discourse parsing due to its vital role in NLP, only a few address nuclearity recognition. Among them only three studies (Li et al., 2015; Chu et al., 2015; Kong and Zhou, 2017) explore nuclearity recognition in Chinese due to the lack of annotated corpus and the abstract nature of Chinese itself. In addition, those studies heavily relied on manual feature engineering (Feng and Hirst, 2014; Heilman and Sagae, 2015; Wang et al., 2017). Only a few studies (Li et al., 2014; Li et al., 2016) used deep neural networks to explore automatic representation learning. One of the disadvantages of previous studies is that they lack deep semantic information extracted from discourse units due to the ineffectiveness of classifier-based models and simple neural network models. Even worse, different from those hypotactic languages such as English, Chinese is a paratactic (discourse-driven and pro-drop) language with a wide spread of ellipsis and open flexible sentence structures. Therefore, the shallow semantic features (syntactic features), which are widely used in English, might not be effective in Chinese. This property makes discourse parsing in Chinese more challenging.

In this paper, we propose a novel text matching network (TMN) for nuclearity recognition. The TMN model encodes the discourse units and paragraphs by combining Bi-LSTM and CNN to capture both the global dependency information and the local n-gram information. Moreover, it introduces three components of text matching, i.e., Cosine, Bilinear and Single Layer Network, to incorporate various similarities and interactions between discourse units and thus provide more useful information to recognise nuclearity. Experimental results on the Chinese Discourse TreeBank (CDTB) (Li et al., 2014) show that our proposed TMN model significantly outperforms various strong baselines in both micro-average and macro-average F1. We summarise the contributions of our work as follows:

- We combine Bi-LSTM and CNN to jointly learn proper representation of the discourse units, which can capture both global dependency information and local n-gram information.
- We introduce three text matching components, i.e., Cosine, Bilinear and Single Layer Network to capture various semantic similarities and interactions among the discourse units.
- We consider the semantic relations between the discourse units and paragraphs. These relations provide an effective supplement to recognise the nuclearity types.

The remainder of this paper is organised as follows: Section 2 introduces the related work, Section 3 gives the details of our model TMN, Section 4 reports the experimental results and Section 5 gives the conclusions.

2 Related Work

Previous studies on nuclearity recognition mainly focused on English, with RST Discourse Treebank (RST-DT) (Carlson et al., 2003) being the most popular corpus. However, most of them only regard

nuclearity recognition as a trivial component of overall discourse parsing, and they ignore its specific characteristics and critical importance.

The algorithms of nuclearity recognition published on RST-DT can mainly be categorised as shift-reduce algorithms (Ji and Eisenstein, 2014; Heilman and Sagae, 2015; Wang et al., 2017), probabilistic CKY-like algorithms (Joty et al., 2013; Li et al., 2014; Li et al., 2016) and greedy bottom-up algorithms (Feng and Hirst, 2014). Li et al. (2016) applied different classifiers to three discourse parsing subtasks separately, but they share the high level representation of discourse units by the same network structures. Wang et al. (2017) used a transition-based system to build discourse trees with nuclearity labels and then used Support Vector Machines (SVMs) to determine the discourse relations at different text levels.

Most of the previous studies used SVMs and variants of Conditional Random Fields (CRFs); only Li et al. (2014) and Li et al. (2016) introduced neural networks into nuclearity recognition. Li et al. (2014) used a two-layer feedforward neural network to determine the relation between text spans and computed the representation for each text span based on the representations of its subtrees by recursive neural models. Li et al. (2016), which is used as one of our baselines, proposed an attention-based hierarchical Bi-LSTM network to learn the representations of the text spans and used a tensor-based transformation function to capture interactions among the features of the text spans.

For recognising nuclearity between Chinese discourse units, there are only three studies. Li et al. (2015), Chu et al. (2015) and Kong and Zhou (2017) have done some preliminary work on Chinese Discourse TreeBank (CDTB) (Li et al., 2014). Li et al. (2015) used contextual features, lexical features and dependency tree features to recognise nuclearity by a Maximum Entropy (ME) classifier. Chu et al. (2015) used similar features to recognise three types of nuclearity by three different ME classifiers and used sampling techniques to obtain a balanced training set and testing set. Kong and Zhou (2017) integrated some previous research and proposed a CDT-styled End-to-End discourse parser, which can automatically detect discourse units and perform all three discourse parsing subtasks in sequence. They used the same model of nuclearity recognition as Li et al. (2015) and reported the same results.

3 Text Matching Network on Nuclearity Recognition

In this section, we propose a novel text matching network (TMN) for nuclearity recognition, and its high-level illustration is shown in Figure 2, which includes three modules: 1) Text Encoding, 2) Text Matching, and 3) Nuclearity Classification.

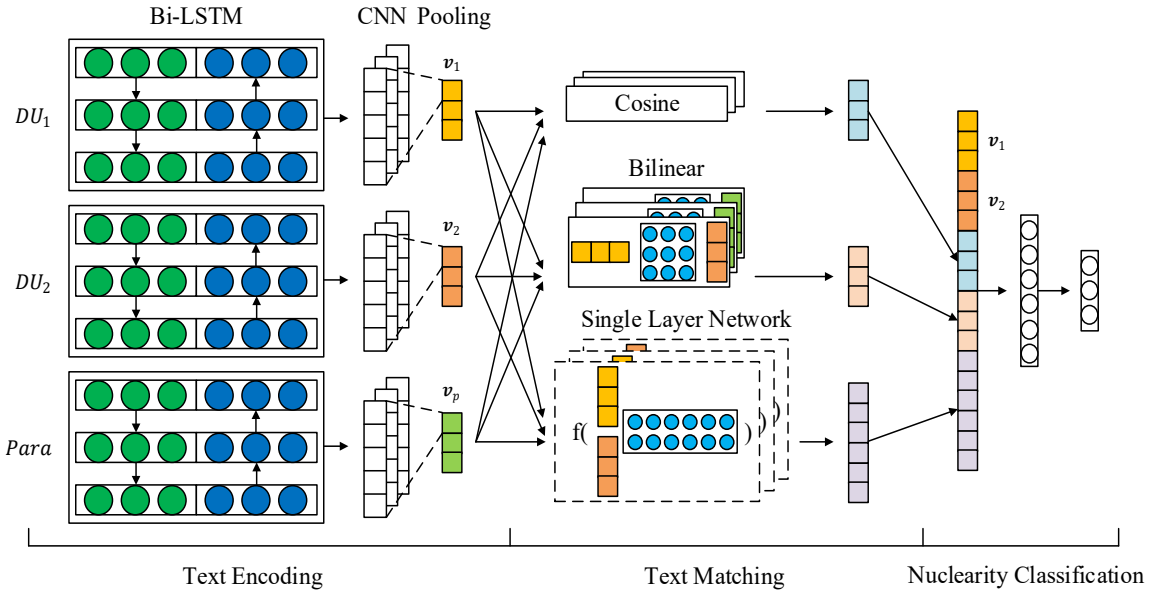

Figure 2: The basic framework of our model, including 1) Text Encoding, 2) Text Matching, and 3) Nuclearity Classification.

To recognise the nuclearity of two discourse units DU_1 and DU_2, their word sequences and the paragraph $Para$ that contains the above two units are the inputs of our model. Taking Example 1 as an instance, DU_1/DU_2 could be one of the EDUs *a*-*g* or their combinations, and $Para$ is the whole paragraph. The Text Encoding module first encodes these word sequences into the semantic vectors $\boldsymbol{v}_1, \boldsymbol{v}_2$ and $\boldsymbol{v}_p$ by Bi-LSTM and CNN. Then, these semantic representations are fed into the Text Matching module, which uses Cosine to calculate the similarity of different semantic vectors and applies Bilinear and Single Layer Network to incorporate the strong linear and nonlinear interactions between different semantic vectors. Finally, the combined feature vector, which is composed of two semantic vectors of two discourse units and one feature vector of all of the interactive information, is sent to the output layer, i.e., the Nuclearity Classification module, through a nonlinear transformation.

Our TMN model is based on two hypotheses. The first hypothesis is that there are strong correlations between the nuclearity and the semantic similarity or interactions of two discourse units. Commonly, the discourse units with similar semantics are multinuclear, and the discourse units with semantic interactions have a mononuclear relation.

The second hypothesis is that the nuclearity of two discourse units is relevant to the topic of the paragraph or document. For example, in the case of a mononuclear relation, the nucleus unit is usually semantically closer to the topic of the paragraph. Therefore, our TMN model makes the semantic match between not only the different discourse units but also the discourse unit and paragraph, by three similarity metrics, namely, the Cosine, Bilinear and Single Layer Network, which can capture the features that are related to nuclearity recognition adequately.

3.1 Text Encoding

Our Text Encoding module combines Bi-LSTM and CNN to encode the discourse unit DU_i and the paragraph $Para$, which is the modification of the Convolutional-pooling LSTM (Tan et al., 2016) in question answering.

Its input is a sequence of words $(t_1, t_2, \dots, t_T)$ in a discourse unit DU_i or a paragraph $Para$ where T is the number of words in the discourse unit or paragraph. Each word t_i in the sequence is represented as the combination of its word embedding $\boldsymbol{e}_i$ and POS (Part-Of-Speech) tag embedding $\boldsymbol{p}_i$ as follows:

$$\boldsymbol{w}_i = [\boldsymbol{e}_i, \boldsymbol{p}_i]. \tag{1}$$

LSTM models successfully keep the useful information from long-range dependency, but they focus more on the words that are behind. Due to the need for our model to treat each word equally, Bi-LSTM is introduced to the Text Encoding module. At each position t, we concatenate the output $\overrightarrow{\boldsymbol{h}_t}, \overleftarrow{\boldsymbol{h}_t} \in \mathbb{R}^l$ of the two inverted LSTMs as the output of the current word as follows:

$$\boldsymbol{h}_t = [\overrightarrow{\boldsymbol{h}_t}, \overleftarrow{\boldsymbol{h}_t}]. \tag{2}$$

Therefore, each output contains not only the current word information but also the contextual information. Consequently, we choose it as the input of a 1D CNN to capture richer local n-gram information. The 1D CNN is similar to the traditional n-gram model. It can effectively capture the local interaction information between the words in the word window, and thus, it can make up for the lack of LSTM. Finally, all of the features captured by the convolution kernels are collected by the global max pooling operation to obtain the textual representation $\boldsymbol{v}_i \in \mathbb{R}^c$. In addition, the number l of LSTM neurons, the size k and the number c of the CNN convolution kernels are all hyperparameters of our model.

3.2 Text Matching

After obtaining the representations $\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_P$ of the discourse units DU_1, DU_2 and the paragraph $Para$ by Bi-LSTM and CNN in the Text Encoding module, we apply the Cosine, Bilinear (Sutskever et al., 2009; Jenatton et al., 2012) and Single Layer Network (Collobert and Weston, 2008) to capture the interactions between different discourse units and between the discourse unit and the paragraph.

The cosine distance calculates the angle between two vectors, which is usually used to measure the degree of similarity. **Cosine** is defined as follows:

$$cos(\boldsymbol{v}_1, \boldsymbol{v}_2) = \frac{\boldsymbol{v}_1^{\top} \boldsymbol{v}_2}{\|\boldsymbol{v}_1\| \cdot \|\boldsymbol{v}_2\|}. \tag{3}$$

Because the discourse units with multinuclear relations are usually similar in content, the cosine similarity $cos(\boldsymbol{v}_1, \boldsymbol{v}_2)$ between the semantic representations of two discourse units can be used as an effective feature to determine their nuclearity. We also calculated the cosine similarities $cos(\boldsymbol{v}_1, \boldsymbol{v}_P)$ and $cos(\boldsymbol{v}_2, \boldsymbol{v}_P)$ between the discourse unit and the paragraph to measure the similarity between the discourse unit and the topic of the paragraph. These two similarities are helpful for identifying the mononuclear relations.

Bilinear is a simple way to incorporate the linear interactions between two vectors and is defined as follows:

$$s(\boldsymbol{v}_1, \boldsymbol{v}_2) = \boldsymbol{v}_1^\top \boldsymbol{W} \boldsymbol{v}_2, \tag{4}$$

where $\boldsymbol{W} \in \mathbb{R}^{c \times c}$ is the parameter matrix. Usually, when using the Bilinear model (Chen et al., 2016; Wan et al., 2016; Wu et al., 2017), the Bilinear value $s(\boldsymbol{h}_{x_i}, \boldsymbol{h}_{y_j}) = \boldsymbol{h}_{x_i}^T \boldsymbol{W} \boldsymbol{h}_{y_j}$ is calculated for any two words in the two word sequences $\{x_1, x_2, \ldots, x_m\}$ and $\{y_1, y_2, \ldots, y_n\}$ to obtain a matching matrix, where $\boldsymbol{h}_{x_i}, \boldsymbol{h}_{y_i}$ are semantic vectors that correspond to word x_i and y_j.

A discourse unit or a paragraph could contain a larger number of words, and it will lead to generating an enormous matching matrix. However, the number of training samples that can be used in our model is relatively small, which results in great difficulty with training the parameter $\boldsymbol{W}$. Therefore, we simplified this process to calculate the Bilinear values $\boldsymbol{v}_1^\top \boldsymbol{W} \boldsymbol{v}_2, \boldsymbol{v}_1^\top \boldsymbol{W} \boldsymbol{v}_P$ and $\boldsymbol{v}_2^\top \boldsymbol{W} \boldsymbol{v}_P$ directly on the encoded discourse units and encoded paragraphs. Since $\boldsymbol{v}_1, \boldsymbol{v}_2$ and $\boldsymbol{v}_P$ contain the semantic information of the discourse units and the paragraph, Bilinear is equivalent to capturing the linear interaction between DU_1, DU_2 and $Para$ at the textual level.

Single Layer Network is defined as follows:

$$\begin{aligned} s(\boldsymbol{v}_1, \boldsymbol{v}_2) &= f(\boldsymbol{V}_1[\boldsymbol{v}_1, \boldsymbol{v}_2] + \boldsymbol{b}_1) \\ s(\boldsymbol{v}_1, \boldsymbol{v}_P) &= f(\boldsymbol{V}_2[\boldsymbol{v}_1, \boldsymbol{v}_P] + \boldsymbol{b}_2), \end{aligned} \tag{5}$$

where $\boldsymbol{V}_1 \in \mathbb{R}^{w \times 2c}, \boldsymbol{b}_1 \in \mathbb{R}^w$ and $\boldsymbol{V}_2 \in \mathbb{R}^{w \times 2c}, \boldsymbol{b}_2 \in \mathbb{R}^w$ are parameters to incorporate nonlinear interactions between the discourse units and between the unit and the paragraph, and we choose $tanh$ as the activation function f. The number w of neurons is the hyperparameter of our model. Because of the existence of nonlinear activation functions, the Single Layer Network can capture nonlinear interactions between different discourse units and between the discourse unit and the paragraph. Bilinear focuses on capturing linear interactions, while Single Layer Network focuses on capturing non-linear interactions, and thus, Single Layer Network can make up for the lack of Bilinear to some extent. We can obtain nonlinear feature vectors $\boldsymbol{v}_{S_{12}}, \boldsymbol{v}_{S_{1P}}$ and $\boldsymbol{v}_{S_{2P}} \in \mathbb{R}^w$ by Single Layer Network as follows:

$$\begin{aligned} \boldsymbol{v}_{S_{12}} &= f(\boldsymbol{V}_1[\boldsymbol{v}_1, \boldsymbol{v}_2] + \boldsymbol{b}_1) \\ \boldsymbol{v}_{S_{1P}} &= f(\boldsymbol{V}_2[\boldsymbol{v}_1, \boldsymbol{v}_P] + \boldsymbol{b}_2) \\ \boldsymbol{v}_{S_{2P}} &= f(\boldsymbol{V}_2[\boldsymbol{v}_2, \boldsymbol{v}_P] + \boldsymbol{b}_2) \end{aligned} \tag{6}$$

With Cosine, Bilinear, and Single Layer Network, we measure the similarity and capture the linear and non-linear interactions among the discourse units and between the unit and the paragraph, and by training the parameter matrices $\boldsymbol{W}, \boldsymbol{V}_1, \boldsymbol{V}_2$ and the parameter vectors $\boldsymbol{b}_1, \boldsymbol{b}_2$, the matching features that play an important role in recognising nuclearity are extracted. This process is how our Text Matching module identifies important information in the discourse unit and performs text matching methods under supervised learning.

3.3 Nuclearity Classification

Based on the discourse unit DU_1, DU_2 and the paragraph $Para$, we obtain the semantic representation vectors $\boldsymbol{v}_1, \boldsymbol{v}_2$ from the Text Encoding module and the Cosine values, the Bilinear values and the nonlinear feature vectors from the Text Matching module. We concatenate all of these values and the vectors above as the input feature vector $\tilde{\boldsymbol{v}}$ of the Nuclearity Classification module as follows:

$$\begin{aligned} \boldsymbol{v}_{cos} &= [\cos(\boldsymbol{v}_1, \boldsymbol{v}_2), \cos(\boldsymbol{v}_1, \boldsymbol{v}_P), \cos(\boldsymbol{v}_2, \boldsymbol{v}_P)]^\top \\ \boldsymbol{v}_{bl} &= [\boldsymbol{v}_1^\top \boldsymbol{W} \boldsymbol{v}_2, \boldsymbol{v}_1^\top \boldsymbol{W} \boldsymbol{v}_P, \boldsymbol{v}_2^\top \boldsymbol{W} \boldsymbol{v}_P]^\top \\ \boldsymbol{v}_{sln} &= [\boldsymbol{v}_{S_{12}}, \boldsymbol{v}_{S_{1P}}, \boldsymbol{v}_{S_{2P}}] \\ \tilde{\boldsymbol{v}} &= [\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_{cos}, \boldsymbol{v}_{bl}, \boldsymbol{v}_{sln}] \end{aligned} \tag{7}$$

We implement the Nuclearity Classification module using a two-layer feedforward neural network. The input vector is first sent to a nonlinear transformation and then fed into a standard softmax layer, where the nonlinear transformation uses the $Relu$ function (Nair and Hinton, 2010) as follows:

$$\boldsymbol{t} = Relu(\boldsymbol{W}_t\,\tilde{\boldsymbol{v}} + \boldsymbol{b}_t) \tag{8}$$

$$\tilde{y} = softmax(\boldsymbol{W}_s\boldsymbol{t} + \boldsymbol{b}_s), \tag{9}$$

where $\boldsymbol{W}_t \in \mathbb{R}^{w_t \times (2c+6+3w)}$, $\boldsymbol{b}_t \in \mathbb{R}^{w_t}$ and $\boldsymbol{W}_s \in \mathbb{R}^{3 \times w_t}$, $\boldsymbol{b}_s \in \mathbb{R}^3$ are the parameters in the nonlinear transformation and in the softmax layer, respectively. Additionally, the number w_t of neurons in the nonlinear transformation layer is the hyperparameter of our model. During the training, we use the Adam optimiser (Kingma and Ba, 2014) to optimise the network parameters by maximising the log-likelihood loss function between the predicted label $\tilde{y}$ and the real label y.

4 Experimentation

In this section, we first introduce the CDTB corpus in the Chinese and experimental setting, and then we report and analyse the experimental results.

4.1 CDTB Corpus

Following the tree structure, the representation of nuclearity in RST and the representation of connectives in the Penn Discourse Treebank (PDTB) (Prasad et al., 2008), Li et al. (2014) built the Chinese Discourse TreeBank (CDTB) corpus based on the Chinese Treebank (CTB) (Xue et al., 2005) with a connective-driven dependency tree scheme. In CDTB, each paragraph is marked as a connective-driven dependency tree (CDT), where its leaf nodes are EDUs, its intermediate nodes represent connectives, and EDUs connected by connectives can be combined into higher level discourse units.

Similar to RST-DT, there are three types of nuclearity in CDTB: Nucleus-Satellite, Satellite-Nucleus and Nucleus-Nucleus. However, CDTB labels relations that link two discourse units on their parent node, and thus, a binary tree with *n* EDUs has only *n-1* relations. Moreover, CDTB marks paragraphs instead of documents as discourse trees, which will also lead to fewer annotated relations.

Currently, the CDTB corpus consists of 500 newswire articles, which are further divided into 2342 paragraphs with a CDT representation for one paragraph. CDTB contains 10650 EDUs, and each EDU has 22 Chinese characters on average. There are 7310 annotated relations in CDTB in which 3555 (48.6%) relations are mononuclear relations, with 2110 Nucleus-Satellite and 1445 Satellite-Nucleus, while the remaining 3755 (51.4%) relations are Nucleus-Nucleus. The reason is that COORDINATION (mononuclear/multinuclear: 468/3683) is the largest category (56.8%) in CDTB, which leads to a large number of multinuclear instances.

4.2 Experimental Setup

We evaluate our model on the corpus CDTB. Following the previous work (Kong and Zhou, 2017), we also choose the same 450 documents as the training set and 50 documents as the testing set. The specific division and labelling situation is shown in Table 1. In our evaluation, all of the non-binary trees are transformed into left binary trees, and the numbers of multinuclear relations in the converted training set and testing set are 4257 and 485, respectively. Moreover, we report the Precision (P), Recall (R) and F1 on each nuclearity type and also give the micro-average and macro-average F1.

	Training set	Testing set
Document	0001-0090, 0101-0190, 0201-0290, 0301-0325, 0400-0454, 0500-0509, 0520-0554, 0590-0596, 0600-0647	0091-0100, 0191-0200, 0291-0300, 0510-0519, 0648-0657
Nuclearity	#Nucleus-Satellite/#Satellite-Nucleus: 1901/1343 # Nucleus-Nucleus: 3371	#Nucleus-Satellite/#Satellite-Nucleus: 207/104 # Nucleus-Nucleus: 384

Table 1: Division of dataset. There are 3244 mononuclear relations and 3371 multinuclear relations in the training set, and these figures are 311 and 384 in the testing set, respectively.

The dimension of the word embeddings is set to 300, and the dimension of the POS embeddings is set to 50. We pre-trained the word embeddings with Word2Vec (Mikolov et al., 2013) on the Wikipedia Chinese corpus[1]. We used HanLP[2] to preprocess the texts, including the word segmentation and POS tagging, and we used the Keras[3] library to implement our model. All of the parameters are randomly initialised except for the word embeddings. We adopted the dropout strategy (Hinton et al., 2012) to avoid overfitting and set the dropout rate to 0.5.

We selected one-ninth of the samples from the training set as a development set to tune the hyperparameters by a grid search, and for a fair comparison, all of the models in our experiment use the same parameters. In the Text Encoding module, the number l of LSTM neurons is set to 50, and the size k and the number c of the CNN convolution kernels are set to 5 and 400, respectively, according to the empirical results in Table 2. In the Text Matching module, the number of neurons w in the Single Layer Network is set to 50. In the Nuclearity Classification module, the number w_t of neurons in the nonlinear transformation layer is set to 128.

Filter size	#Feature map				
	100	200	300	400	500
3	55.06	57.73	58.34	58.18	59.38
4	56.47	57.23	58.42	59.21	59.66
5	56.84	57.71	59.17	**59.70**	58.44
6	56.65	58.85	58.22	58.53	57.05

Table 2: Macro-average F1 with different CNN parameter settings on the development set.

4.3 Experimental Results

To exhibit the effectiveness of our TMN model, the experiment results consist of two parts: the baselines and TMN.

Baselines: We collect five baselines for our experiment: ME (Kong and Zhou, 2017), Bi-LSTM, Bi-LSTM(A), Bi-LSTM+CNN and Bi-LSTM(A)+T (Li et al., 2016). The ME model proposed by (Kong and Zhou, 2017) used contextual features, lexical features and dependency tree features to recognise nuclearity by an ME classifier. We obtained their source codes and found a data partition error in their system: some instances appeared in both the training set and the testing set. For a fair comparison, we corrected the data partition following their paper and report the revised results of their system in this paper. Considering that the attention-based hierarchical Bi-LSTM network proposed by (Li et al., 2016) performs better than the recursive neural model (Li et al., 2014), we implemented four neural network-based systems. The first is a Bi-LSTM network model (Bi-LSTM), and the second is a Bi-LSTM network model with the attention mechanisms (Bi-LSTM(A)). The third is a Bi-LSTM network model with the attention mechanisms and the tensor-based transformation function (Bi-LSTM(A)+T) (Li et al., 2016). The fourth is a Bi-LSTM+CNN model that combines Bi-LSTM and CNN.

Model	Nucleus-Satellite P / R / F1	Satellite-Nucleus P / R / F1	Nucleus-Nucleus P / R / F1	Macro-F1	Micro-F1
ME	32.2 / 15.1 / 20.5	40.0 / 15.0 / 21.8	65.6 / **87.8** / 75.0	42.3	60.5
Bi-LSTM	53.6 / 50.2 / 51.9	30.4 / 33.7 / 32.0	74.3 / 74.6 / 74.5	52.8	62.9
Bi-LSTM(A)	55.7 / 44.9 / 49.7	34.9 / 36.5 / 35.7	74.6 / 80.0 / 77.2	54.4	65.2
Bi-LSTM+CNN	59.6 / 46.4 / 52.1	**40.2** / 31.7 / 35.5	73.2 / 83.5 / 78.0	55.7	67.1
Bi-LSTM(A)+T	56.8 / **50.7** / 53.6	37.5 / 43.4 / 40.2	**77.0** / 77.9 / 77.5	57.2	66.3
TMN	**69.1** / 45.4 / **54.8**	39.2 / **49.0** / **43.6**	76.2 / 83.3 / **79.6**	**60.4**	**69.0**

Table 3: The experimental results of five baselines and TMN.

The experimental results of the above models are shown in Table 3, and these results show that our

[1] https://dumps.wikimedia.org/zhwiki/
[2] https://github.com/hankcs/HanLP
[3] https://keras.io/

TMN model outperforms the other five baselines in both the micro-average and macro-average F1. Compared with the traditional method ME, the other five neural network models improve the micro-average and macro-average F1 significantly, especially the macro-average F1, with large gains from 10.5 up to 18.1. These results justify the effectiveness of the neural network models on the nuclearity recognition to capture the deeper semantic information that is hiding in the discourse units.

Compared with Bi-LSTM, Bi-LSTM(A) improves the macro-average and micro-average F1 by 1.6 and 2.3, respectively, because Bi-LSTM(A) can pick up prominent semantic information on the output of Bi-LSTM using the attention mechanism. Furthermore, the performance of Bi-LSTM(A)+T is better than Bi-LSTM(A), and this result ensures that tensor-based transforms are also helpful to Bi-LSTM. Moreover, due to the Bi-LSTM+CNN model combining the ability of capturing the global information by Bi-LSTM and the local information by CNN, it performs better than both Bi-LSTM and Bi-LSTM(A).

Our TMN model outperforms all of the other five models, with large gains from 3.2 up to 18.1 in the macro-average F1 and a significant gain from 1.9 up to 8.5 in the micro-average F1. Compared with the Bi-LSTM, Bi-LSTM(A) and Bi-LSTM+CNN, which focus on obtaining the representations of text, our TMN model can capture the semantic features and incorporates interactions between the representations of the discourse units and the paragraphs. Moreover, compared with the Bi-LSTM(A)+T model, our TMN combines both Bi-LSTM and CNN in the Text Encoding module and uses many simple but efficient methods to incorporate richer interactions in the Text Matching module.

4.4 Analysis

We also compare the performances of the different nuclearity types, and Table 3 shows that the performance of multinuclear (Nucleus-Nucleus) is much higher (>24 in F1) than that of the mononuclear relations (Nucleus-Satellite and Satellite-Nucleus). This result derives from two aspects. The first is that the majority of the training set is Nucleus-Nucleus, which occupies 56.8% of all annotated nuclearity, while the percentages of the Nucleus-Satellite and Satellite-Nucleus are 25.3% and 17.9%, respectively. The second is that our Text Matching module is helpful for identifying similar discourse units via the matching mechanisms of Cosine, Bilinear and Single Layer Network, and then assigns Nucleus-Nucleus to them.

The Bi-LSTM+CNN model is a simplified version of TMN that does not use the Text Matching module. Compared with the Bi-LSTM+CNN model, TMN combines the semantic similarity and the interaction information simultaneously to improve the macro-average and micro-average F1 by 4.7 and 1.9, respectively. These figures justify our first hypothesis that there are strong correlations between the nuclearity and the semantic similarity or the interactions of two different discourse units.

To analyse the contribution of each mechanism in the Text Matching module, we conduct experiments on some variants of TMN, and the results are shown in Table 4. In Table 4, the basic model (TMN-CBS) is equal to Bi-LSTM+CNN, which does not have the Text Matching module. The TMN-BS model refers to the TMN model whose Text Matching module only uses Cosine, while TMN-C refers to the TMN model whose Text Matching module only uses Bilinear and Single Layer Networks.

Model	Nucleus-Satellite P / R / F1	Satellite-Nucleus P / R / F1	Nucleus-Nucleus P / R / F1	Macro-F1	Micro-F1
TMN-CBS	59.6 / 46.4 / 52.1	40.2 / 31.7 / 35.5	73.2 / 83.5 / 78.0	55.7	67.1
TMN-BS	56.5 / 50.7 / 53.4	**47.6** / 28.9 / 35.9	74.2 / **83.7** / 78.7	56.8	68.0
TMN-C	61.5 / **54.1** / **57.6**	39.6 / 34.6 / 36.9	75.7 / 81.7 / 78.6	57.8	68.3
TMN-P	60.9 / 51.2 / 55.6	34.7 / 41.4 / 37.7	**76.9** / 79.0 / 77.9	57.3	66.8
TMN	**69.1** / 45.4 / 54.8	39.2 / **49.0** / **43.6**	76.2 / 83.3 / **79.6**	**60.4**	**69.0**

Table 4: Experimental results of variants of the TMN Model.

Compared with TMN-CBS, the TMN-BS model and the TMN-C model improve the macro-average and micro-average F1, and these improvements show that the semantic similarity or interaction information captured by the Cosine, Bilinear, and Single Layer Network are helpful for nuclearity recognition. Especially after adding the interaction information using the Bilinear and Single Layer Network, the F1 of the relation Nucleus-Satellite achieves a 5.5% improvement.

The TMN-P model is a simplification of the TMN model, which removes the input of the paragraph *Para* in Figure 2. TMN-P captures only the similarity and interaction information between the different discourse units, without the similarity and interaction information between the discourse unit and the paragraph. Compared with TMN-P, TMN significantly improves the macro-average and micro-average F1 by 3.1 and 2.2, respectively, which justifies our second hypothesis that nuclearity recognition is relevant to the topic of the paragraph.

Nuclearity	Nucleus-Satellite	Satellite-Nucleus	Nucleus-Nucleus
Nucleus-Satellite	-	14.5%	40.1%
Satellite-Nucleus	9.6%	-	41.4%
Nucleus-Nucleus	6.6%	10.1%	-

Table 5: The percentages of misclassified samples.

Table 5 shows the error statistics of our TMN model in nuclearity recognition. It shows that 40.1% of the Nucleus-Satellite instances and 41.4% of the Satellite-Nucleus instances are frequently identified as Nucleus-Nucleus by our TMN model. These results show that the errors mainly arise from the judgment of whether an instance is mononuclear or multinuclear. This finding is mainly due to two reasons: 1) the number of Nucleus-Nucleus instances accounts for more than half of the training set; and 2) many discourse units differ in nuclearity but are semantically similar. We consider the following two discourse units as examples.

Example 2: 农业获得较好收成 $_{a}$，全年粮食总产量达七十六点六亿公斤 $_{b}$。 *Farming received good harvests* $_{a}$, *the total grain output in the year amounted to 7.66 billion kg* $_{b}$.

The nuclearity type between the two EDUs *a* and *b* in Example 2 is Nucleus-Satellite due to the content in EDU *a* being more generalised and being able to semantically contain the content described in EDU *b*. However, there is a strong correlation between "农业 *farming*" and "粮食 *grain*" and between "收成 *harvest*" and "产量 *output*", at a semantic level. Therefore, the Text Matching module will misjudge their relation as Nucleus-Nucleus via the similarity and interaction information between the two EDUs *a* and *b*.

5 Conclusions

In this paper, we propose a novel TMN model for nuclearity recognition in Chinese discourse. First, we employ a Text Encoding module to capture both the global dependency information and the local n-gram information via Bi-LSTM and CNN. In this way, the overall discourse semantics can be much better represented. Then, we employ a Text Matching module to capture various similarities and interactions between different discourse units and between the encoded unit and the paragraph by the Cosine, Bilinear and Single Layer Network. Here, while Cosine calculates the semantic similarity, Bilinear and Single Layer Network incorporate the strong linear and nonlinear interactions between the semantic vectors. Experimental results on the CDTB corpus show that our TMN model significantly outperforms various strong baselines both in micro-average and macro-average F1. Our future work will focus on how to better tune the input of our neural network model and apply this model to other languages.

Acknowledgements

The authors would like to thank three anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China under Grant Nos. 61772354, 61773276 and 61673290, and was also supported by the Strategic Pioneer Research Projects of Defense Science and Technology under Grant No. 17-ZLXDXX-02-06-02-04.

Reference

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*. Springer, Dordrecht, pages 85-112.

Jifan Chen, Qi Zhang, Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Implicit discourse relation detection via a deep architecture with gated relevance network. In *ACL 2016.* pages 1726-1735.

Xiaomin Chu, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. 2015. Recognizing nuclearity between chinese discourse units. In *IALP 2015*. IEEE, pages 197-200.

Ronan Collobert, and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML 2008*. ACM, pages 160-167.

Vanessa Wei Feng, and Graeme Hirst. 2014. A linear-time bottom-up discourse parser with constraints and post-editing. In *ACL 2014*. pages 511-521.

Michael Heilman, and Kenji Sagae. 2015. Fast rhetorical structure theory discourse parsing. *arXiv preprint arXiv:1505.02425*.

Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *Computer Science,* 3(4):212-223.

Mikel Iruskieta, Arantza Díaz de Ilarraza, and Mikel Lersundi. 2014. The annotation of the central unit in rhetorical structure trees: A key step in annotating rhetorical relations. In *COLING 2014*. pages 466-475.

Rodolphe Jenatton, Nicolas L. Roux, Antoine Bordes, and Guillaume R. Obozinski. 2012. A latent factor model for highly multi-relational data. In *NIPS 2012*. Curran Associates Inc, pages 3167-3175.

Yangfeng Ji, and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *ACL 2014*. pages 13-24.

Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Yashar Mehdad. 2013. Combining intra-and multi-sentential rhetorical parsing for document-level discourse analysis. In *ACL 2013*. pages 486-496.

Diederik P. Kingma, and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Fang Kong, and Guodong Zhou. 2017. A CDT-styled end-to-end Chinese discourse parser. In *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP 2017)*. 16(4):26.

Jiwei Li, Rumeng Li, and Eduard Hovy. 2014. Recursive deep models for discourse parsing. In *EMNLP 2014*. pages 2061-2069.

Qi Li, Tianshi Li, and Baobao Chang. 2016. Discourse parsing with attention-based hierarchical neural networks. In *EMNLP 2016*. pages 362-371.

Yancui Li, Fang Kong, and Guodong Zhou. 2014. Building Chinese discourse corpus with connective-driven dependency tree structure. In *EMNLP 2014*. pages 2105-2114.

Yancui Li, Jing Sun, Wenhe Feng, Guodong Zhou. 2015. The platform of Chinese discourse structure analysis based on connective-driven dependency tree. In *China National Conference on Computational Linguistics (CCL 2015)*.

Annie Louis, Aravind Joshi, and Ani Nenkova. 2010. Discourse indicators for content selection in summarization. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, pages 147-156.

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.

Daniel Marcu. 2000. *The theory and practice of discourse parsing and summarization*. MIT press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*.

Vinod Nair, and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *ICML 2010*. Omnipress, pages 807-814.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K. Joshi, and Bonnie L. Webber. 2008. The Penn Discourse Treebank 2.0. In *LREC 2008*. pages 2961-2968.

Manfred Stede. 2008. RST revisited: Disentangling nuclearity. '*Subordination' versus 'Coordination' in Sentence and Text: A cross-linguistic perspective*. pages 33-59.

Ilya Sutskever, Joshua B. Tenenbaum, and Ruslan R. Salakhutdinov. 2009. Modelling relational data using Bayesian clustered tensor factorization. In *NIPS 2009*. pages 1821-1828.

Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2016. Improved representation learning for question answer matching. In *ACL 2016*. pages 464-473.

Suzan Verberne, Lou Boves, Nelleke Oostdijk, and Peter-Arno Coppen. 2007. Evaluating discourse-based answer extraction for why-question answering. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pages 735-736.

Yizhong Wang, Sujian Li, and Houfeng Wang. 2017. A two-stage parsing method for text-level discourse analysis. In *ACL 2017*. pages184-188.

Shengxian Wan, Yanyan Lan, Jiafeng Guo, Jun Xu, Liang Pang, and Xueqi Cheng. 2016. A deep architecture for semantic matching with multiple positional sentence representations. In *AAAI 2016*. AAAI Press, pages 2835-2841.

Yu Wu, Wei Wu, Chen Xing, Zhoujun Li, and Ming Zhou. 2017. Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots. In *ACL 2017*. pages 496-505.

Naiwen Xue, Fei Xia, Fudong Chiou, and Marta Palmer. 2005. The Penn Chinese Treebank: Phrase structure annotation of a large corpus. *Natural language engineering* 11(2): 207-238.

Bowei Zou, Guodong Zhou, and Qiaoming Zhu. 2014. Negation focus identification with contextual discourse information. In *ACL 2014*. pages 522-53.