Memory Augmented Attention Model for Chinese Implicit Discourse Relation Recognition

Yang Liu, Jiajun Zhang and Chengqing Zong

Institute of automation, Chinese Academy of Sciences
{yang.liu2013,jjzhang,cqzong}@nlpr.ia.ac.cn

Abstract Recently, Chinese implicit discourse relation recognition has attracted more and more attention, since it is crucial to understand the Chinese discourse text. In this paper, we propose a novel memory augmented attention model which represents the arguments using an attention-based neural network and preserves the crucial information with an external memory network which captures each discourse relation clustering structure to support the relation inference. Extensive experiments demonstrate that our proposed model can achieve the new state-of-the-art results on Chinese Discourse Treebank. We further leverage network visualization to show why our attention and memory model are effective.

Keywords: Chinese Implicit Relation Recognition; Memory Agumented Neural Network; Attention Neural Model

1 Introduction

The Chinese implicit discourse relation recognition has drawn more and more attention, because it is crucial for Chinese discourse understanding. Recently, the Chinese Discourse Treebank (CDTB) was released [1]. Although Chinese Discourse corpora shares the similar annotation framework with Penn Discourse Treebank (PDTB) for English, the statistical differences are obvious and significant. First, the connectives in Chinese occur much less frequently than those in English [2]. Second, the relation distribution in Chinese is more unbalanced than that in English. Third, the relation annotation for Chinese implicit case is more semantic due to the language essential characteristic [3]. These evidences indicate that implicit discourse relation recognition task for Chinese would be different from English.

Unfortunately, there is existing few work on Chinese discourse relation problem [4, 7], thus our work is mainly inspired by the studies of English. Conventional approaches on identifying English discourse relation rely on handcrafted features extracted from two arguments, including word-pairs [8], VerbNet classes [10], brown clustering [24], production rules[15] and dependency rules [9]. These features indeed capture the correlation with discourse relation to some extent and achieve considerable performance in explicit cases. However, implicit discourse relation recognition is much harder, due to the absence of connectives[1]. Moreover, these hand-crafted features usually suffer

[1] The connective has strong correlation with discourse relations

from data sparsity problem [19] and are weak to capture the deep semantic feature of discourse [22].

To tackle this problem, deep learning methods are introduced to this area. It can learn dense real-valued vector representations of the arguments, which can capture the semantics in some extent, and alleviate the data sparsity problem simultaneously. Recently, a variety of neural network architectures have been explored on this task, such as convolution neural network [32], recursive network [22], feed-forward network [26], recurrent network [25], attentional network [23] and hybrid feature model[6, 5]. These studies show that deep learning technology can achieve comparable or even better performance than the conventional approach with complicated hand-crafted features.

More recently, there are growing interest in memory augmented neural architecture. The advantage of extra memory is to capture and preserve useful information for task, the core of this idea is to keep those information in independent memory slot, and trigger and retrieval the related memory slot to support the inference. This design has proven effective in many works, including neural turing machine [17], memory network [28], dynamic memory networks [21], matching networks [29], etc.

Therefore, in this paper, we propose a memory augmented attention model (MAAM) to handle Chinese implicit discourse relation recognition task. It can represent arguments with an attention-based neural network, and then retrieval the external memory for relation inference support information, after that it combines the representation and memory support information to complete the classification.

More specifically, the procedure of our model can be divided into five steps: **1)** Our model use a general encoder module to transform the input arguments from word sequence into dense vectors. **2)** An attention module is proposed to score the importance of each word based on the given contexts and the weighted sum of the words is used as the argument representation. **3)** An external memory is employed to produce an output based on this arguments representation. **4)** The memory gate combines the memory output together with the attention representation to generate a refined representation of the arguments. **5)** Finally, we stack a feed-forward network as the classification layer to predict the discourse relation. Extensive experiments and analysis show that our proposed method achieves the new state-of-the-art results on Chinese Discourse Treebank (CDTB).

2 Memory Augmented Attention Model

In this section, we first give an overview of the modules that build up memory augmented attention model (MAAM). We then introduce each module in detail and give intuitions about its formulation. A high-level illustration of the MAAM is shown in Fig.1.

As shown in Fig.1, our framework consists of five modules: 1) general encoder module; 2) content-based attention module; 3) external memory module; 4) memory gate; 5) classification module.

The **General Encoder Module** encodes the word sequence of the two arguments into distributed vector representations. It is implemented by using the bidirectional recurrent neural network.

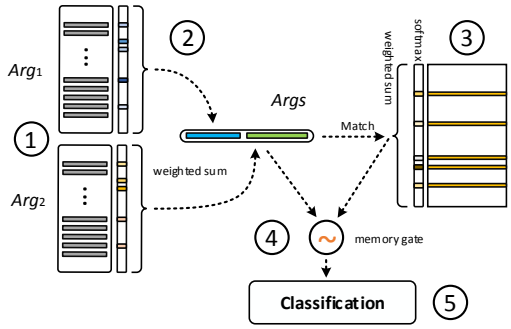

Fig. 1. The basic framework of our model, including 1) General Encoder Module, 2) Content-based Attention Module, 3) External Memory Module, 4) Memory Gate and 5) Classification Module.

The **Attention Module** is proposed to capture the importance (attention) of each word in two arguments. We score the weight of each word in the argument based on its inner context and generates a weighted sum as the argument representation.

The **External Memory Module** consists of a fixed number of memory slots. The external memory computes the match score between the representation of arguments and yields a probability distribution. Then memory generates a weighted sum as memory output.

The **Memory Gate** is a learn-able controller component and it computes the convex combination of the original argument representation and the memory output to generate a refined representation.

The **Classification Module** stacks on the refined representation of the arguments and outputs the final discourse relation. We implement this module with a two-layer feed-forward network which can capture the interaction between two arguments implicitly.

2.1 General Encoder Module

In implicit discourse relation recognition, the input is the word sequence of two arguments Arg1 and Arg2. We choose recurrent neural network [16] to encode the arguments. Word embeddings are given as input to the recurrent network. At each time step *t*, the network updates its hidden state $h_t = RNN(x_t, h_{t-1})$, where x_t is the embedding vector of the *t*-th word of the input argument. In our model, we use a gated recurrent unit (GRU) to replace the normal RNN unit [12]. GRU is a variant of RNN, which works much better than the original one and suffers less from the vanishing gradient problem by introducing the gate structure like Long Short Term Memory (LSTM) [18]. Assume each time step *t* has an input x_t and a hidden state h_t. The formula of

GRU shows as follows:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{1}$$
$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{2}$$
$$\tilde{h_t} = tanh(W x_t + r_t \circ U h_{t-1} + b_h) \tag{3}$$
$$h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \tilde{h_t} \tag{4}$$

In brief, the simple version of GRU is $h_t = GRU(x_t; h_{t-1})$. RNN and its variant as described above read an input sequence *x* in order, starting from the first word to the last one. However, we expect the representation of each word to summarize not only the preceding words, but also the following words. Thus, we propose to use a bidirectional RNN [27]. A Bi-RNN consists of a forward and a backward RNN. The forward RNN reads the input sequence from left to right, while the backward RNN reads the sequence in the reverse order.

$$\overrightarrow{h_t} = \overrightarrow{GRU}(x_t, \overrightarrow{h_{t-1}}) \tag{5}$$
$$\overleftarrow{h_t} = \overleftarrow{GRU}(x_t, \overleftarrow{h_{t-1}}) \tag{6}$$

We obtain representation for each word by concatenating two hidden state sequences generated by the forward and backward RNNs.

$$h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \tag{7}$$

In this way, the representation h_t of each word contains the summary of both the preceding words and the following words.

2.2 Attention Module

After obtaining the representation of the arguments by treating each word equally in general encoder module, we now apply the content-based attention module to score the importance of each word in the arguments. We evaluate the weight of each word only based on the its inner context. The motivation behind it is that since the connective is absent in implicit samples, we can utilize the context of the arguments to generate an appropriate representation. Obviously, the contribution of each word in the context is not same and it is natural to capture the correlation between the context dependent word feature and the discourse relation using attention mechanism. In our case, we use a multilayer perception to implement the attention module:

$$e_t = u_a^T tanh(W_a h_t + b_a) \tag{8}$$

Notice that h_t is generated by the general encoder module. The weight of each word h_t is computed using softmax:

$$a_t = \frac{exp(e_t)}{\sum_{j=1}^{T} exp(e_j)} \tag{9}$$

For instance, we consider the vector v_{Arg1} the weighted sum of the representations of Arg1:

$$v_{Arg1} = \sum_{j=1}^{T} a_t h_t \tag{10}$$

We generate the vector of Arg2 in the same way. Then we directly concatenate two vectors as the representation of arguments:

$$v_{Args} = [v_{Arg1}; v_{Arg2}] \tag{11}$$

2.3 External Memory Module

As long as we have the semantic representation of arguments, we can use it to interact with our augmented memory. Our external memory consists of the memory slots, which are activated by the particular pattern of the arguments and generate corresponding output as response. This memory output will be used in following step to refine the original argument representation. Concretely, we first compute the similarity score between v_{Args} and each memory slot m_i and produce a normalized weight w_i using similarity measure $K[\cdot, \cdot]$. Also, in order to improve the focus, a sharpen factor β is needed.

$$w_i \leftarrow \frac{exp(\beta K[v_{Args}, m_i])}{\sum_j exp(\beta K[v_{Args}, m_j]))} \tag{12}$$

In our case, we use the cosine similarity as our metric.

$$K[u, v] = \frac{u \cdot v}{\|u\| \cdot \|v\|} \tag{13}$$

Then, we generate the output from memory according to the weights.

$$m = \sum_i w_i m_i \tag{14}$$

The memory design is mainly inspired by Neural Turing Machine[17]. The memory will capture the common pattern of discourse relation distribution during training. For example, when an input relation sample accesses the external memory, the memory will response with an output vector which contains the information mostly related to the similar samples it has seen before. Intuitively, samples with similar representations usually belong to the same discourse relation. In summary, the memory actually implicitly holds the discourse relation clustering information for the following classification. The external memory component is randomly initialized and optimized during training.

2.4 Memory Gate

Once we can access the output information m from memory, we can use it to generate the refined representation $\widetilde{v}$ along with the original representation of arguments v_{Args}.

We propose an interpolation strategy to combine these two vectors together and employ a sigmoid function called memory gate to control the final output.

$$\alpha = \sigma(W_g[v_{Args}; m] + b_g) \tag{15}$$

Where σ is a sigmoid function. We then compute a convex combination of the memory output and the original argument representation:

$$\widetilde{v} = \alpha \cdot v_{Args} + (1 - \alpha) \cdot m \tag{16}$$

The memory gate is a learn-able neural layer. The idea behind it is that although memory can return the clustering structure information which is potentially useful. Also, we build a gate mechanism to control the output of memory and mix them with the original argument representations.

2.5 Classification Module

Given the refined representation vector $\widetilde{v}$ of the arguments, we implement the classification module using a two-layer feed-forward network which is followed by a standard softmax layer.

$$\tilde{y} = softmax(tanh(W_c\widetilde{v} + b_c)) \tag{17}$$

Where $\tilde{y}$ is our output predicted label. During training, we optimize the network parameters by maximizing the cross-entropy loss function between the true and predicted labels.

3 Experiments

3.1 Corpora

We evaluate our model on Chinese Discourse Treebank (CDTB) [1–3, 25], which has been published as standard corpora in CoNLL shared task 2016. In our work, we experiment on the ten relations in this corpus following the setup of suggestions given by the shared task. We directly adopt the standard training set, development set, test set and blind test set. We also use the word embeddings provided by the CoNLL 2016.

3.2 Training Details

To train our model, the objective function is defined as the cross-entropy loss between the outputs of the softmax layer and the ground-truth class labels. We use adadelta algorithm to optimize the whole neural networks. To avoid over-fitting, dropout operation is applied on the layer before softmax.

3.3 Experimental Results

To exhibit the effectiveness of our model, our experiment results consists of three parts: baselines, MAAM variants and MAAMs.

Baselines: We collect two baselines for our experiments, the one is "Conjunction" and another is "Focused RNN" which achieved the ***best*** result in CoNLL 2016 shared task.

We implement the first "Conjunction" system which directly annotates every test sample as "Conjunction". The reason behind is that due to the unbalanced problem of corpora (see Table 1), this baseline system is very strong according to the CoNLL report by Xue et al [25] and many participated systems cannot beat this baseline.

The "Focused RNN" is proposed by Weiss and Bajec [31], which is implemented with a focused recurrent neural network which can selective react to different context. Its result is directly selected from the report of CoNLL 2016.

MAAM variants: Since there are few published results on CDTB, it is necessary to show variants of our model. These variants are helpful to understand the contribution of each module, since the variants we proposed is only sightly different from our final model. The detail of each MAAM variants is shown below.

MAAM+0memslot+no Encoder: It use no encoder module at all. In this variant, it directly uses word embedding sequence to encode arguments, and applies the same attention layer on them. This model explores the effectiveness of embedding features missing context dependent information.

MAAM+0memslot+GRU encoder: This system only uses single GRU as the encoder of input module, it is used to understand the effectiveness of bidirectional encoder.

MAAM+0memslot+Mean(no Attention): Instead of using attention mechanism, this system directly represent argument as mean of all hidden states in Bi-GRU, treating each word in argument equally.

We can see from Table 2 that the proposed MAAM module is better than all the variants. It is obvious that both of the context and the attention are beneficial for distributed argument representation in discourse relation.

System		**Development**	**Test**	**Blind Test**
Baseline	Conjunction	61.96	63.59	68.14
	Focused RNN (2016, Best Result)	**66.67**	**64.07**	**70.68**
MAAM Variants	MAAM+0memslot+no Encoder	66.63	64.18	70.62
	MAAM+0memslot+GRU Encoder	66.67	65.01	71.62
	MAAM+0memslot+Mean(no Attention)	66.23	64.01	70.45
MAAMs	MAAM+0memslot	66.87	65.03	71.89
	MAAM+1memslot	67.00	65.02	72.10
	MAAM+20memslots	**67.54**	**66.02**	**73.16**
	MAAM+50memslots	68.43	65.92	72.77
	MAAM+100memslots	68.20	65.73	72.56
	MAAM+150memslots	67.44	65.08	72.38

Table 1. The Experiment results on CoNLL 2016 Shared Task

MAAMs: Now, we compare our memory augmented attention model (MAAM) with other approaches in the closed track. Our memory models (containing different numbers of slots [1,20,50,100,150] can outperform the two baselines, and the one with 20 slots achieves the best result, which is the new state-of-the-art on CTDB. Specifically, we observe an interesting phenomenon in our memory models. Along with the number of memory slots grow, the performance is improved first (from 0 to 20 slots) but is gradually decreased (from 20 to 50, 100 and 150). We speculate that the underfitting problem (no adequate training samples) is the main reason. When comparing to MAAM+0memslot, we can see that all the settings of memory model can obtain better results, demonstrating the effectiveness of proposed external memory component.

3.4 Discussion and Analysis

The experimental results demonstrate the superiority of our memory augmented attention model. In this section, we discuss the behavior of the external memory and the attention module in the network.

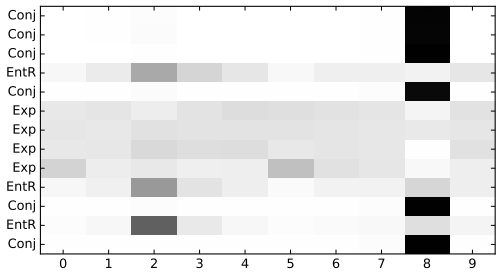

Fig. 2. Memory activation for different relation samples. Horizontal coordinate reflects the activation of 10 memory slots. Vertical coordinate reflects different discourse implicit relation samples. (Conjunction-Conj; Expansion-Exp; EntRel-EntR) Each row in figure represent the different activation of different memory slot for each input discourse relation sample. The deeper color indicate higher score.

Memory Analysis: The results show that the external memory component is significantly helpful for the performance. In order to understand how our memory component works, we show a memory component which contains 10 memory slots in Fig.2. As we mentioned above, the memory slot will be triggered when the relevant input arguments retrial the memory component. The memory will compute scores for each memory slot based on input arguments, we call these scores as activation. We now feed 13 arguments belong to different discourse relation into memory component. The activations of each 10 memory slots triggered by different relation samples are shown in Fig.2, the deeper color means this slot achieve higher activation, each row in Fig.2 exhibits the different activation of memory slot for every input relation arguments. As we can see that, arguments belong to the same relation always trigger the same slots (location) in

memory component. For instance, the ”EntRel” samples always focus on the 2-nd slot (in horizontal) and the ”Conjunction” samples trigger the 8-th slot.

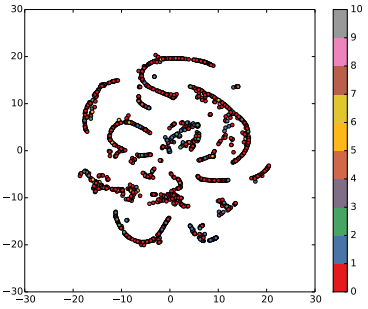

Fig. 3. t-SNE for Chinese discourse relation distribution. Notice that clustering for each relation in figure. The “Expansion” is in blue. Conjunction-0; Expansion-1; EntRel-2; AltLex-3; Causation-4; Contrast-5; Purpose-6; Conditional-7; Temporal-8; Progression-9. As we can see, the “Conjunction” relation plays as a background for the rest of relations.

Representation Analysis: In order to understand the discourse relation distribution (representation) in our model, we show the t-SNE visualization of Chinese implicit discourse relation samples in Fig.3 (using feature space from classification module). As we can see, the ”Conjunction” relation samples mostly play as a background for any other relation. This may be caused by the definition of “conjunction”.[2] Meanwhile, other relation samples are hard to distinguish from ”Conjunction” samples. This situation also indicates that the Chinese implicit relation recognition is a difficult task.

Attention Analysis: Our attention module scores each word relying on the inner content. It captures the correlation between content and discourse relation, different from independent word embedding information which can not access the surrounding context. In Fig 4, the ”Causation” relation example extracted from corpora shows our model pays more attention on the content words than the function words. We annotated the alignment relation between the Chinese relation sample and its English translation. The attention module focuses on the ”*international*;*steady*;*expansion*” in Arg1 and ”*for China's export*;*provides*;*international environment*” in Arg2, which can be roughly considered as a simple summarization of two arguments. This example demonstrates the effect of the proposed attention module. The result of attention makes us to wonder if we should give different score to word when we deal with different relation.

Discussion: Another issue we observed is the ambiguity and data imbalance of Chinese implicit discourse relation. Comparing to English, Chinese contains more less explicit connectives, this is the main reason for Chinese implicit reason recognition problem. Therefore, many relation samples is hard distinguish from ”Conjunction”, unless it is pretty obvious for annotator. Our approach is actually based on a assumption

[2] Conjunction: relation between two equal-status statements serving a common communicative function, *from CoNLL 2016*. It is relative ambiguous.

Fig. 4. Attention for *Causation* samples. The attention module focus on the "international,steady,expansion" in Arg1 and "for China's export,provides,international environment" in Arg2.

that every relation has a *prototype* sample, thus we hope our memory component can capture each discourse relation prototype and identify it from unseen sample. However, we didn't observed positive result to support our assumption.

4 Related Work

Implicit discourse relation recognition has been a hot topic in recent years. However, most of the approaches focus on English. There are mainly two directions related to our study: 1) English implicit discourse relation recognition using neural networks, and 2) memory augmented networks.

Conventional implicit relation recognition approaches rely on kinds of hand-crafted features [11, 8, 24], these surface features usually suffer from sparsity problem. Then, neural network based approaches are proposed. In order to alleviate feature sparsity problem, Ji&Eisenstein [19] first transform surface features of arguments into low dimension distributed representations to boost the performance. A discourse document usually covers different scale unit from word, sentence to paragraph. To model this kind of structures, Li [22] and Ji [20] both introduced the recursive network to represent arguments to facilitate the discourse parsing.Considering the discourse relation recognition as text classification problem, Liu et al [23] propose a convolution neural network (CNN) to detect the sequence feature in arguments to predict relation. Rutherford et al [25] conduct experiments to explore the effectiveness of feedforward neural network and recurrent neural network. Liu&Li [23] use attention mechanism to refine the representation of arguments by reweighing the importance of different parts of argument. Braud and Denis [13, 14] utilize the word representation to improve implicit discourse relation classification. Their method investigates the correlation between word embedding and discourse relation.

The memory model is inspired by recently proposed memory augmented network. The Neural Turing Machine (NTM) [17] builds an external memory component to preserve kinds of subsequence pattern explicitly, and makes NTM more effective to learn from training samples. Another type of memory augmented network is memory network [28], which is different from NTM and works more like a cache for particular data. The memory network saves the sentences in memory to support multiple step question&answer inference. More recently, the matching network is proposed by Vinyals et al [29], its memory component caches the common pattern of representation and corresponding label of training samples. It predicts label by matching input sample with

memory caches then generate weighted sum label (with matching distribution) as final output. Since the memory network can capture particular pattern of samples and be optimized during training, we extend it in our framework to maintain crucial information for Chinese implicit relation recognition. The experimental results verify the efficacy of the proposed memory network and the memory augmented model achieves the best performance on CDTB.

5 Conclusion

In this paper, we have proposed a memory augmented attention model for Chinese implicit Discourse relation recognition. The attention network is employed to learn the semantic representation of the two arguments Arg1 and Arg2. The memory network is introduced to capture the underlying clustering structure of samples. The extensive experiments show that our proposed method achieves the new state-of-the-art results on CDTB.

Acknowledgments The research work has been supported by the Natural Science Foundation of China under Grant No. 61333018 and No. 61403379.

References

1. Nianwen Xue: Annotating discourse connectives in the Chinese treebank. In Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky:84-91 (2005)
2. Yuping Zhou and Nianwen Xue: Pdtb-style discourse annotation of Chinese text. ACL2012:69-77 (2012)
3. Yuping Zhou and Nianwen Xue:The Chinese Discourse Treebank: A Chinese corpus annotated with discourse relations. Language Resources and Evaluation 49(2):397-431 (2015)
4. Mei Tu, Yu Zhou and Chengqing Zong: Automatically parsing Chinese discourse based on maximum entropy. Acta Scientiarum Naturalium Universitatis Pekinensis 50(1):125-132 (2014)
5. Haoran Li, Jiajun Zhang and Chengqing Zong: Implicit Discourse Relation Recognition for English and Chinese with Multi-view Modeling and Effective Representation Learning. ACM Transactions on Asian and Low-Resource Language Information Processing 16(3):19 (2017)
6. Haoran Li, Jiajun Zhang, Yu Zhou and Chengqing Zong: Predicting Implicit Discourse Relation with Multi-view Modeling and Effective Representation Learning. NLPCC:374-386 (2016).
7. Yancui Li, Wenhe Feng, Jing Sun, Fang Kong, and Guodong Zhou: Building Chinese discourse corpus with connective-driven dependency tree structure. In EMNLP:2105-2114 (2014)
8. Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng: Recognizing implicit discourse relations in the penn discourse treebank. In EMNLP:343-351 (2009)
9. Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan: A pdtb-styled end-to-end discourse parser. Natural Language Engineering 20(02):151–184 (2014)
10. Xiaomian Kang, Haoran Li, Long Zhou, Jiajun Zhang, and Chengqing Zong: An end-to-end Chinese discourse parser with adaptation to explicit and nonexplicit relation recognition. ACL-Conll share task:27-32 (2016)

11. Emily Pitler and Ani Nenkova: Using syntax to disambiguate explicit discourse connectives in text. In ACL-IJCNLP:13-16 (2009)
12. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
13. Chlo Braud and Pascal Denis: Comparing word representations for implicit discourse relation classification. In EMNLP (2015)
14. Chlo Braud and Pascal Denis: Learning connective-based word representations for implicit discourse relation identification. In EMNLP:203-213. (2016)
15. Prashant Chandrasekar, Xuan Zhang, Saurabh Chakravarty, Arijit Ray, John Krulick, and Alla Rozovskaya: The virginia tech system at conll-2016 shared task on shallow discourse parsing. CoNLL Shared Task.2016: 115-121 (2016)
16. Jeffrey L Elman: Distributed representations simple recurrent networks, and grammatical structure. Machine learning 7(2-3):195–225 (1991).
17. Alex Graves, Greg Wayne, and Ivo Danihelka: Neural turing machines. arXiv preprint arXiv:1410.5401 (2014)
18. Sepp Hochreiter and Jurgen Schmidhuber: Long short-term memory. Neural computation 9(8):1735–1780 (1997)
19. Yangfeng Ji and Jacob Eisenstein: Representation learning for text-level discourse parsing. In ACL:13-24 (2014)
20. Yangfeng Ji and Jacob Eisenstein: One vector is not enough: Entity-augmented distributed semantics for discourse relations. TACL (3):329–344 (2015)
21. Ankit Kumar, Ozan Irsoy, Jonathan Su, James Bradbury, Robert English, Brian Pierce, Peter Ondruska, Ishaan Gulrajani, and Richard Socher: Ask me anything: Dynamic memory networks for natural language processing. CoRR, abs/1506.07285 (2015)
22. Jiwei Li, Rumeng Li, and Eduard H Hovy: Recursive deep models for discourse parsing. In EMNLP:2016-2069 (2014)
23. Yang Liu and Sujian Li: Recognizing implicit discourse relations via repeated reading: Neural networks with multi-level attention. In EMNLP:1224-1233 (2016)
24. Attapol Rutherford and Nianwen Xue: Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. In EACL. vol 645 (2014)
25. Attapol T Rutherford, Vera Demberg, and Nianwen Xue: Neural network models for implicit discourse relation classification in English and Chinese without surface features. arXiv preprint arXiv:1606.01990 (2016)
26. Niko Schenk, Christian Chiarcos, Kathrin Donandt, Samuel Ronnqvist, Evgeny A Stepanov, and Giuseppe Riccardi: Do we really need all those rich linguistic features? a neural network-based approach to implicit sense labeling. ACL:41-50 (2016)
27. Mike Schuster and Kuldip K Paliwal: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing 45(11) (1997)
28. Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al: End-to-end memory networks. In NIPS:2440-2448 (2015)
29. Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra: Matching networks for one shot learning. NIPS 2016:3630-3638 (2016)
30. Jianxiang Wang and Man Lan: Two end-to-end shallow discourse parsers for English and Chinese in conll-2016 shared task. CoNLL-2016 shared task:33-40 (2016)
31. Gregor Weiss and Marko Bajec: Discourse sense classification from scratch using focused rnns. ACL2016 1(100):50 (2016)
32. Nianwen Xue, Hwee Tou Ng, Attapol Rutherford, Bonnie Webber, Chuan Wang, and Hongmin Wang: Conll 2016 shared task on multilingual shallow discourse parsing. Proceedings of the CoNLL-16 shared task:1-19 (2016)