Implicit Discourse Relation Classification Based on Semantic Graph Attention Networks

Yuhao Ma
Capital Normal University
Information Engineering College,
Beijing 100089 China
mayuhao1013@163.com

Yu Yan
Capital Normal University
Information Engineering College,
Beijing 100089 China
15698488809@163.com

Jie Liu
Capital Normal University
Information Engineering College,
Beijing 100089 China
liujxxxy@126.com

ABSTRACT

The<fn id="fn5" fn-type="other">
</fn>implicit discourse relation classification is of great importance to discourse analysis. It aims to identify the logical relation between sentence pair. Compared with the linear network model, the graph neural network has a more complex structure to capture cross-sentence interactions. Therefore, this article proposes a semantic graph neural network for implicit discourse relation classification. Specifically, we design a semantic graph to describe the syntactic structure of sentences and semantic interactions between sentence pair. Then, convolutional neural network (CNN) with different convolutional kernels to extract the multi-granularity semantic features. The experimental results on Penn Discourse TreeBank 2.0 (PDTB 2.0) prove that our work performed well.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial intelligence**; • **Natural language processing**; • **Discourse, dialogue and pragmatics**;

KEYWORDS

Implicit discourse relation classification, Discourse parsing, Semantic interaction, Graph attention network

ACM Reference Format:
Yuhao Ma, Yu Yan, and Jie Liu. 2021. Implicit Discourse Relation Classification Based on Semantic Graph Attention Networks. In *The 5th International Conference on Computer Science and Application Engineering (CSAE 2021), October 19–21, 2021, Sanya, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3487075.3487156

1 INTRODUCTION

Sentences in discourse do not exist in isolation, but are connected by logical relations. The purpose of implicit discourse relation classification is to recognize the logical relationship between sentences, it is the basic research of discourse parsing and helpful for many natural language processing tasks, such as machine translation [1], question-answering system [2], text summarization [3]. As shown in Figure 1, according to whether a complex sentence contains connectives, discourse relations can be divided into two types: the explicit discourse relationship with connectives in the sentence and the implicit discourse relationship without connectives. The research on explicit discourse relation classification has achieved good results, with a score of f1 exceeding 93%. For the implicit discourse relation classification, due to the lack of shallow linguistic features, inferential discourse relations can only rely on the deep semantic information. As a result, this task is still the bottleneck of discourse parsing.

Explicit: *That hung over parts of the factory,* ***even though*** *exhaust fans ventilated the area.*
Implicit: *The ploy worked. The defense won.*

Figure 1: Examples of Discourse Relations, Explicit Instances Have Connectives, and Implicit Instances do not Have Connectives.

A lot of research has been done on implicit discourse relation classification. Traditional methods of machine learning mainly used artificially designed language features [4], but the discourse relations are deeply rooted in semantics, and they are hard to identify from shallow features. With the development of deep learning, people used neural networks to learn better textual representations. Unlike traditional sentence modeling, implicit discourse relation recognition is a double sequence problem, and the direct link between sentences is expected to play an important role. Some studies used the attention mechanism [5], gating mechanism [6] to model semantic interaction, and achieved some results, but how to conduct deep semantic interaction and capture more effective information is still a challenge. Recently, models based on the graph neural network, such as the graph attention network [7], have attracted widespread attention. Compared with linear network models, the graph attention network has a more complex structure to capture cross-sentence interactions. In many downstream tasks, the pre-trained model showed strong abilities, from the latest research [8] results of implicit discourse relation classification, we found a trend of cooperation between the pre-trained model and relation classification.

In summary, this article proposes a new semantic graph neural network to solve the problem of relationship classification. First, we use BERT to encode each sentence into word-embedding that combines contextual information. Second, we design a new strategy to model the semantic interaction, and learn the dependency relationship of sentences and the features of the interaction between

CSAE 2021, October 19–21, 2021, Sanya, China

ACM ISBN 978-1-4503-8985-3/21/10…$15.00
https://doi.org/10.1145/3487075.3487156

sentences through the graph attention network. Then, CNN captures the N-gram information with different granularities through a plurality of convolution filters with different sizes. After the pooling layer, the significant features are aggregated and sent to the MLP classifier.

2 MODEL

2.1 Model Overview

Figure 2 shows the framework of our model. It consists of five parts: a word embedding layer that uses BERT [9] to map each word with contextual representations, an interaction layer that models the deep semantic interaction between discourse units, a dynamic convolution layer that uses various sizes of convolutional filters, automatically extracts the multi-granularity features, an aggregation layer that removes redundant information and converts features into vectors with a fixed length, a prediction layer that MLP classifier for calculating the probability distribution of the discourse relationship.

2.2 Word Embedding Layer

The pre-trained model is the latest development in the area of deep learning, like Elmo [10], BERT. Different from traditional word embedding, the pre-trained model can generate word embedding combined with the contextual sentence, this article uses BERT to obtain sentence representations. Let $\langle P = \{p_1, p_2, \ldots, p_M\}, Q = \{q_1, q_2, \ldots, q_N\}\rangle$ be an arbitrary sentence pair, where p_i represents the i-th word in sentence P and q_i represents the i-th word in sentence Q. First, we use the Byte-Pair Encoding in BERT to encode the sentences and obtain the following representations:

$$\begin{array}{l} sentence_1 : \left[CLS, e_1^1, e_2^1, \ldots\ldots e_M^1, EOS\right], \\ sentence_2 : \left[CLS, e_1^2, e_2^2, \ldots\ldots e_N^2, EOS\right], \end{array} \tag{1}$$

where M is the length of $sentence_1$ and N is the length of $sentence_2$, and e_k^i is the word-level embedding of the k^{th} word in $sentence_i$, CLS and EOS are special token embeddings in BERT, so the lengths of $sentence_1$ and $sentence_2$ respectively are $M+2$ and $N+2$. We concatenate representations of $sentence_1$ and $sentence_2$ as follows:

$$\begin{array}{l} e_0, e_1, e_2, \ldots . e_M, e_{1+M}, \ldots., e_{M+N+2}, e_{M+N+3} \\ = \left[CLS, e_1^1, \ldots . e_M^1, SEP, SEP, e_1^2, \ldots e_N^2, EOS\right] \end{array} \tag{2}$$

SEP is a special token embedding, used to indicate the boundaries of the sentence connection, inspired by [11], two SEP_s are used here, because we will split these embeddings into two sequences for interaction and fusion in the next few sections.

After several transformed layers, two sentence representations with contextual are obtained $[h_0, h_1, \ldots., h_{M+N+2}, h_{M+N+3}]$, then separate the representations of the two parameters:

$$\left[h_0^1, h_1^1, \ldots., h_{M+1}^1\right] = [h_0, h_1, \ldots., h_{M+1}] \tag{3}$$

$$\left[h_0^2, h_1^2, \ldots., h_{N+1}^2\right] = [h_{M+2}, h_{M+3}, \ldots., h_{M+N+3}] \tag{4}$$

2.3 Interaction Layer

Graph attention network (GAT) was first proposed by [7], it can process complex structured data and update the representation of nodes by gathering information from neighbor nodes, and implicitly

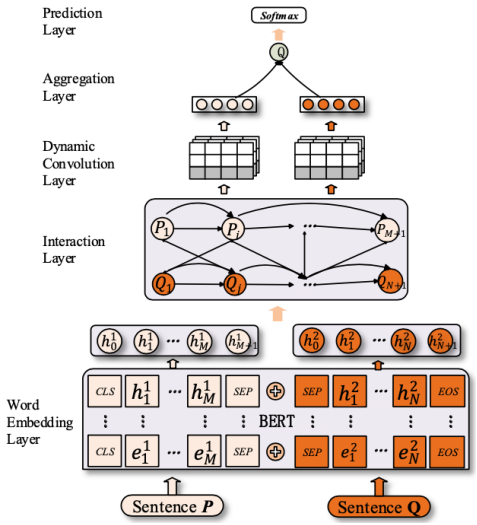

Figure 2: Model Overview.

assign different weights to different nodes in the neighborhood. Given the graph $\mathcal{G}(v, \varepsilon)$, $\mathcal{V} = \{v_i\}$ represents graph nodes set, $h_i^{(0)}$ is the initial feature matrix, where each row represents an initial input feature for a node. In GAT learning, the representation $h_i^{(L)}$ of the hidden layer is obtained by encoding the graph structure and the node features, L is the number of GAT layers. $\mathcal{E} = \{e_{ij}\}$ is set of graph edges. We model the semantic interaction with sentence pair $\langle P, Q\rangle$, take words as nodes, and create two types of edges: inside edges and interactive edges, Figure 3 gives an example of deep semantic interaction.

Inside edges, used to connect nodes in the same sentence and describe the dependencies within the sentence. This article uses the StandfordNLP to perform syntactic parsing and semantic role tagging on two sentences, respectively, explicitly injecting linear dependencies.

Interactive edges, connect the nodes of different sentences and combine the two-sentence graphs into one graph. We consider a maximum interaction strategy that connects any word in one sentence with any word in another sentence, but this creates a large number of redundant edges. The sequential context describes the linguistic features of local co-occurrence (between words) and has been widely used in text representation learning, this article uses the sliding window strategy to describe this sequential context information with point-wise mutual information and calculate the interactive correlation degree of each pair of words. If the semantic association exceeds the predefined threshold of $\alpha_{co-occurs}$ the two words are related in sentence pair. The degree of correlation of each word pair can be obtained by the following calculation:

$$d\left(p_i, q_j\right) = \log \frac{f\left(p_i, q_j\right)}{f\left(p_i\right) f\left(q_j\right)} \tag{5}$$

Where $f(p_i, q_j)$ is the probability of word pair (p_i, q_j) co-occurring in the dame sliding window, which is always estimated by $\frac{\#N_{co-occurs}(p_i, q_j)}{\#N_{windows}}$, where $\#N_{windows}$ is the total number sliding windows over the whole text corpus and $\#N_{co-occurs}(p_i, q_j)$ is the

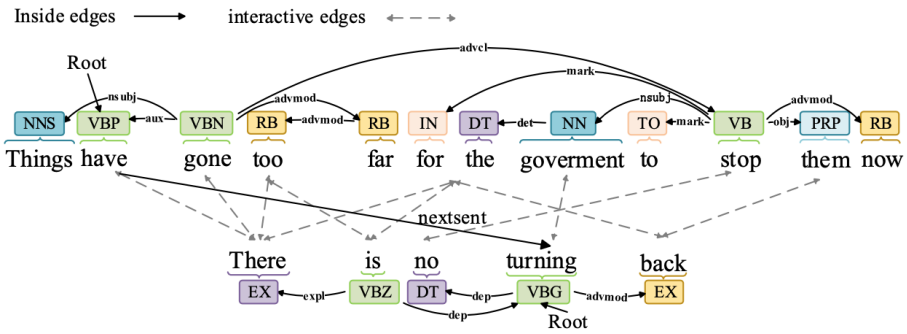

Figure 3: An Example of Semantic Interaction. The Inside Edges Represent the Dependency Relationship, and the Interactive Edges Represent the Degree of Semantic Association between Words and the Connection between Sentence Roots.

number of times that the word pair (p_i, q_j) co-occurs in the same sliding windows over the whole text corpus.

Using these two kinds of edges, the graph attention network can simultaneously learn the dependency of sentences and the interactive features between sentence pair. Let $H^{(L)}=\{h_0^{(L)}, h_1^{(L)}, \ldots, h_{M+N+2}^{(L)}, h_{M+N+3}^{(L)}\}$ be the hidden states of nodes in L-th GAT layer and it is designed as:

$$z_{ij} = LeakyRule\left(W_a\left[Wh_i^{(L)}, Wh_j^{(L)}\right]\right) \tag{6}$$

$$a_{ij} = softmax_j\left(z_{ij}\right) = \frac{exp\left(z_{ij}\right)}{\sum_{k\in\mathcal{N}_i} exp\left(z_{ik}\right)} \tag{7}$$

$$h_i^{(L+1)} = tanh\left(\sum_{j\in\mathcal{N}_i} a_{ij} W h_j^{(L)}\right) \tag{8}$$

Where $\mathcal{N}_i$ stands for a neighborhood of node i, a_{ij} represents the attention weight from node i to neighbor node j. W_a, W are trainable parameterized weight matrix for the attention mechanism. To stabilize the learning process, this article uses a multi-head attention mechanism.

$$h_i^{(L+1)} = tanh\left(\sum_{K=1}^{K}\sum_{j\in\mathcal{N}_i} a_{ij}^k W^K h_j^{(L)}\right) \tag{9}$$

a_{ij}^k is the normalized attention coefficient calculated by the attention coefficient the k-th time. W^K is trainable parameterized weight matrix for the attention mechanism.

2.4 Dynamic Convolution Layer

Inspired by [12], this article uses CNN to extract unigram, bigram, . . ., n-gram information with different granularities by setting different convolutional filtering kernel sizes. Let $m_i^1 \in \mathbb{R}^{(M+1)\times d}$ and $m_i^2 \in \mathbb{R}^{(N+1)\times d}$ are the output features of sentences P and Q through GAT respectively. We apply the convolutional operations to $[m_i^1, \ldots, m_{M+1}^1]$, then the output of each operation applied on arguments is:

$$o_c^1 = ReLU(Conv_c(\left[m_i^1, \ldots, m_{i+c}^1\right] \tag{10}$$

Where the kernel size of $Conv_c$ is c, the stride is 1, and the dimension of o_c^1 depended on the number of filters. As before, we can obtain o_c^2 similarly.

2.5 Aggregation Layer

At this level, this article uses Max-pooling and Attentional-pooling [13] on o_c^1 and o_c^2 respectively. Max-pooling considers that the largest feature is more representative, so only the largest feature is taken as the retained value of the extracted features in a certain dimension, and all other features are discarded. Attentional-pooling may prevent the loss of some important features. By using these two pooling operations, the model can obtain the crucial information on different parts in the parameters.

$$\text{Max} - \text{pooling} : \mathrm{O}_{max}^1 = max_{i=0}^{M+1} o_c^1, \mathrm{O}_{max}^2 = max_{j=0}^{N+1} o_c^2 \tag{11}$$

$$Attentional - pooling : u_c^1 = W_1 \tan\left(W_2 o_c^1\right) \tag{12}$$

$$a_c^1 = \exp\left(u_c^1\right) / \sum_{k}^{M+1} u_c^1 \tag{13}$$

$$\mathrm{O}_{attn}^1 = \sum_{i=1}^{M+1} a_c^1 o_c^1 \tag{14}$$

Where M is the sentence length, W_1, W_2 are trainable matrix parameters, It is possible to obtain O_{attn}^2 via same operations. Then, the vectors are concatenated after pooling operations in order to obtain vectors with a fixed length:

$$Q = \left[\mathrm{O}_{max}^1; \mathrm{O}_{attn}^1; \mathrm{O}_{max}^2; \mathrm{O}_{attn}^2\right] \tag{15}$$

2.6 Prediction Layer

Finally, the vector Q obtained from the aggregation layer is sent to a feed-forward neural network to obtain more abstract representations, and the softmax function is applied to calculate the prediction probabilities of all classes. To estimate the distribution of the data, we trained the end-to-end model by minimizing the loss of cross-entropy between the output of the prediction layer and the actual relationship.

$$\mathcal{L}\left(\hat{y}, y\right) = -\sum_{i=1}^{K} y_i log\left(P_r\left(\hat{y}_i\right)\right) \tag{16}$$

Where $P_r(\hat{y}_i)$ is the predicted probability of the i-th label, K is the number of relation classes.

Table 1: The Statistics of Top-Level Classes in PDTB 2.0

Relation	Train	Val.	Test
Comparison	1896	195	152
Contingency	3236	293	276
Expansion	6994	661	558
Temporal	692	63	75
Total	**12845**	**1212**	**1061**

Table 2: Comparisons of F1 Scores (%) and Accuracy (%) for 4-Class Classification on the Top Classes.

Model	F1	Acc
[16]	47.80	57.39
[17]	48.82	57.44
[18]	50.20	59.13
[19]	52.19	60.69
[20]	52.89	59.66
[21]	58.48	65.26
Our work	63.32	67.08

3 EXPERIMENTS

This section will evaluate the efficacy of this method through experiments. First, we will introduce the PDTB 2.0 [14], then describe the experimental settings, and finally give the experimental results and analysis.

3.1 Datasets

PDTB 2.0 is the largest manually annotated corpus of discourse relations currently available, which has been annotated with 2,159 articles in the Wall Street Journal. PDTB 2.0 has three levels of senses, including classes, types, and subtypes. Our experiments were conducted on four top-level classes, namely, Contingency, Comparison, Temporal, and Expansion, and adopted two experimental settings, binary classification, and multi-class classification. Following the setting [15], we divided the corpus into training set (Section 2-20), validation set (Section 0-1), and test set (Section 21-22). The statistics of the top-level discourse classes are shown in Table 1

3.2 Training

BERT is used to output word embedding with a hidden state size of 768 dimensions. The maximum length of the input sequence is set at 512 and the minimum length is 3. This article uses StanfordNLP as the dependency parser. For the graph attention network, we set the number of layers at 2 and the number of attention heads at 4 with the hidden size of 64. Adam is used to perform gradient optimization on parameters training, the batch size is set at 8, iteration number is 100, the learning rate is set at 0.001, the dropout rate is 0.2. The super parameter $\alpha_{co-occurs}$is set at 0.4. The model was implemented by Pytorch, and all experiments were performed on 2 NVIDIA 2080Ti GPU.

3.3 Results and Analysis

Table 2 shows the results of our model and previous work on 4-class classification of top-level classes. Our model achieves good results in all classification settings, proving its effectiveness. Different from these methods, this article considers wider contexts, model the semantic interaction of the sentence structure, and extract the deeper interaction features from the learned semantic structure, which is the main reason why our method is better than the previous work.

Table 3 shows the results of our model and previous work on the binary classification of top-level classes. Temporal has a lower F1 score because it has the smallest number of samples in the corpus, but our model achieves better results compared with the method using external data to expand the training set. The comparison and extension also performed better, probably for the following reasons: First, self-attention in the encoding process provided valid context information. Second, some discourse units have similar word pairs. The multi-head attention mechanism captures the information of sentence structure and combines the extracted sentence features to generate cross-correlation features, which enables CNN to effectively capture different aspects of the cross-correlation features and capture more useful information from n-gram with different granularities.

4 CONCLUSIONS

This article proposes a semantic graph attention network model for implicit discourse relation classification, which models sentences from the syntactic structure and learns the features of deep-level semantic interaction. As far as we know, we are the first to use the graph neural network to solve this task. Experiments conducted on PDTB 2.0 show that our model can compete with other models. In the subsequent work, the problem of data sparseness and how to learn information about long-term syntactic dependence are two issues to be solved. We will consider using external knowledge to

Table 3: Binary Classification F1 Scores (%) on Top Classes.

Model	Comp.	Cont.	Exp.	Temp.
[16]	40.73	58.96	72.47	38.50
[17]	46.79	57.09	70.41	45.61
[18]	44.10	56.02	72.11	44.41
[19]	47.15	55.24	70.82	38.20
[20]	45.34	51.80	68.50	45.93
Our work	**56.62**	**59.66**	**73.95**	**46.43**

improve the efficiency of our model and extend this approach to different languages.

ACKNOWLEDGMENTS

This work is supported by National Science and technology innovation 2030 major projects (2020AAA0109700), National Natural Science Foundation of China (62076167), and Beijing Municipal Education Commission-Beijing Natural Fund Joint Funding Project (KZ201910028039).

REFERENCES

[1] J. L Li, M. carpuat, and A. Nenkova (2014). Assessing the discourse factors that influence the quality of machine translation. Association for Computational Linguistics, Baltimore, Maryland, USA, 283-388.

[2] P. Jansen, M. Surdeanu, and P. Clark (2014). Discourse complements lexical semantics for non-factoid answer reranking. Association for Computational Linguistics, ACL, Baltimore, Maryland, USA, 977-986.

[3] A. Cohan, F. Dernoncourt, S. Kim, W. Chang, and N. Goharian (2018). A discourse-aware attention model for abstractive summarization of long documents. arXiv preprint arXiv:1804.05685.

[4] E. Pitler, A. Louis, A. Nenkova (2009). Automatic sense prediction for implicit discourse relations in text. Association for Computational Linguistics and International Joint Conference on Natural Language Processing, Suntec, Singapore, 683-691.

[5] Y. Liu, S. Li (2016). Recognizing implicit discourse relations via repeated reading: neural networks with multi-level attention. Empirical Methods in Natural Language Processing, Austin, Texas, USA, 1224-1233.

[6] J. Chen, Q. Zhang, P. Liu, X. Qiu, and X. Huang (2016). Implicit discourse relation detection via a deep architecture with gated relevance network, Association for Computational Linguistics, Berlin, Germany, 1726-1735.

[7] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio (2018). Graph attention networks. International Conference on Learning Representations, Vancouver, BC, Canada.

[8] W. Shi, V. Demberg (2019). Next sentence prediction helps implicit discourse relation classification within and across domains. Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing, Hong Kong, China, 5790-5796.

[9] J. Devlin, M.W. Chang, K Lee, and K Toutanova (2019). BERT: pre-training of deep bidirectional transformers for language understanding. North American Chapter of the Association for Computational Linguistics, Minneapolis, Minnesota, USA, 4171-4186.

[10] M. Peters, M. Neumann, M. Iyyer. M. Gardner, et, al. (2018). Deep contextualized word representations. North American Chapter of the Association for Computational Linguistics, Minneapolis, Minnesota, USA, 2227-2237.

[11] X. Liu, J. Ou, Y. Song, and X. Jiang (2020). On the importance of word and sentence representation learning in implicit discourse relation classification. proceedings of International Joint Conference on Artificial Intelligence, Yokohama, Japan.

[12] K. Sun, Y. Li, D. Deng, and Y. Li (2019). Mulit-Channel CNN based inner-attention for compound sentence relation classification. IEEE Access. 14801-14809.

[13] R. Girdhar, D. Ramanan (2017). Attentional pooling for action recognition. Proceeding of Advances in Neural Information Processing Systems, Long Beach, CA, USA.

[14] R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, and B. Webber (2008). The penn discourse TreeBank 2.0. International Conference on Language Resources and Evaluation, 24, 2961–2968.

[15] Y. Ji, J. Eisenstein (2015). One vector is not enough: Entity-augmented distributed semantics for discourse relations. Transactions of the Association for Computational Linguistics. Beijing, China, 329-344.

[16] M, Lan, J. Wang, Y. Wu, Z. Niu, and H. Wang (2017). Multi-task attention-based neural networks for implicit discourse relationship representation and identification. Proceedings of Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 1299-1308.

[17] Z. Dai, R. Huang (2018). Improving discourse relation classification by modeling inter-dependencies of discourse units in a paragraph. Proceedings of North American Chapter of the Association for Computational Linguistics, New Orleans, Louisiana, USA, 141-151.

[18] S. Varia, C. Hidey, T. Chakrabarty (2019). Discourse relation prediction: revisiting word pairs with convolutional networks. SIGdial Meeting on Discourse and Dialogue, Stockholm, Sweden, 442-452.

[19] H Bai, H Zhao, J Zhao (2019). Memorizing all for implicit discourse relation recognition. arXiv preprint arXiv:1908.11317.

[20] Z. Dai, R. Huang (2019). A regularization approach for incorporating event knowledge and coreference relations into neural discourse parsing. Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing. Hong Kong, China, 2976-2987.

[21] Y. Kishimoto, Y. Murawaki, S. Kurohashi (2020). Adapting BERT to implicit discourse relation classification with a focus on discourse connectives. Conference on Language Resources and Evaluation, Marseille, France, 1152-1158.