Predicting Implicit Discourse Relation with Multi-view Modeling and Effective Representation Learning

Haoran Li, Jiajun Zhang, Yu Zhou and Chengqing Zong
National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
{haoran.li, jjzhang, yzhou, cqzong}@nlpr.ia.ac.cn

Abstract. Discourse relations between two text segments play an important role in many natural language processing (NLP) tasks. The connectives strongly indicate the sense of discourse relations, while in fact, there are no connectives in a large proportion of discourse relations, i.e., implicit discourse relations. The key for implicit relation prediction is to correctly model the semantics of the two discourse arguments as well as the contextual interaction between them. To achieve this goal, we propose a multi-view framework that consists of two hierarchies. The first one is the model hierarchy and we propose a neural network based method considering different views. The second one is the feature hierarchy and we learn multi-level distributed representations. We have conducted experiments on the standard benchmark dataset and the results show that compared with several methods our proposed method can achieve the best performance in most cases.

1 Introduction

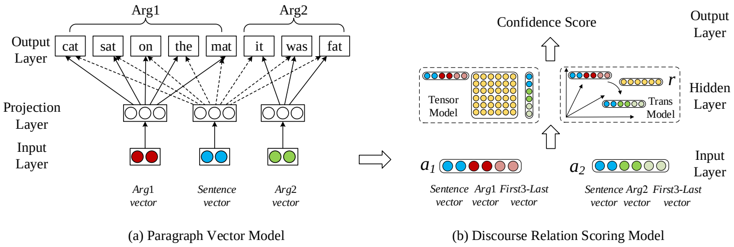
Discourse relation inference is a pivotal task for discourse analysis. According to whether there are connectives or not, discourse relations can be categorized into explicit and implicit relations. The goal of our task is to recognize implicit discourse relations existing between two given discourse arguments. Most of the existing work regards this task as a classification problem and typical classifier such as SVM are leveraged to perform this task. Recently, neural networks are employed to boost the recognition performance.

Besides the model, the features are also important. The previous work usually resorts to discrete features [11, 15, 8, 10, 1, 13] which strongly depend on the linguistic lexicons and lead to data sparsity. Recently, many studies [5, 3, 24, 18] demonstrate that the distributed representations can improve implicit relation prediction. However, the distributed representations focus only on certain aspects, such as surface words or arguments, which lack modeling multi-level features.

In this paper, we propose a multi-view framework to tackle implicit relation recognition and the architecture of our model is shown in Figure 1. Our framework consists of two hierarchies. One is the model hierarchy (Figure 1(b)) which is based on neural network modeling discourse relations from the relation classification view and the relation transformation view. The other is the feature hierarchy (Figure 1(a)) which learns distributed representations from different levels, namely from words, arguments, syntactic structures to sentences.

We make the following contributions in this paper.

- We design a multi-view neural network model to recognize implicit discourse relations considering the interactions between two discourse arguments and the relation transformation property.
- We propose to represent discourse arguments by multi-level distributed features, from words, arguments, syntactic structures to sentences.



(a) Paragraph Vector Model

(b) Discourse Relation Scoring Model

Fig. 1. The architecture of our model. To simplify the figure we neglect the word vectors of the input layer in the Paragraph Vector Model.

Table 1. Implicit Discourse Relation Examples in PDTB

1	Sense	Comparison.Contrast
	Arg1	The common view is that there will be mild economic growth, modest profit expansion, and things are going to be hunky-dory.
	Arg2	Our view is that we may see a profit decline.
2	Sense	Expansion.Instantiation
	Arg1	Futures prices declined.
	Arg2	One below 50 indicates a contraction may be ahead.

2 Overview of the Penn Discourse Treebank

The PDTB [16] is the largest available English discourse corpus. The senses of discourse relations are organized into a hierarchical structure in which the top level contains four major classes: Comparison, Contingency, Expansion and Temporal. The next two levels consist of more fine-grained relation types. A discourse relation instance consists of two arguments denoted by $Arg1$ and $Arg2$. We give two examples in PDTB in Table 1.

3 Model

3.1 Discourse Relation Scoring Model

As shown in Figure 1(b), each discourse relation argument pair is represented as two dense embeddings a_1 and $a_2 \in \mathbb{R}^{H_1}$ where H_1 is the size of the embeddings. The representation learning of a_1 and a_2 will be introduced in Section 4. Then a_1 and a_2 serve as input of the neural network model in which we design multiple kinds of hidden layers. Above the hidden layer, the model outputs a confidence score for specific relations using a linear transformation of the following function:

$$f(a_1, a_2) = W^T h$$

$h \in \mathbb{R}^{H_2}$ denotes hidden layer representation and $W \in \mathbb{R}^{H_2}$ denotes the linear transformation vector . H_2 is the size of the hidden layer.

To better investigate the hidden layer h, we apply multiple types of networks: Single-Layer Neural Network, Tensor Neural Network and Transformation (Trans) Neural Network.

Single-Layer Model. This model is the simplest form of neural network containing only one hidden layer. It is defined as follows:

$$h = tanh(W_s[a_1; a_2] + b_s)$$

where $W_s \in \mathbb{R}^{H_2 \times 2H_1}$ and $b_s \in \mathbb{R}^{H_2}$. $[a_1; a_2] \in \mathbb{R}^{2H_1}$ denotes concatenation of a_1 and a_2.

Tensor Model. A tensor is a multi-dimensional array that can connect two input vectors in every dimension. Tensor model has been widely used in many NLP tasks [20, 14]. It can be defined as follows:

$$h = tanh(a_1^T {W_t}^{[1:H_2]} a_2 + W_s[a_1; a_2] + b_s)$$

where $W_t^{[1:H_2]} \in \mathbb{R}^{H_1 \times H_1 \times H_2}$ is a H_2-way tensor.

Tensor model can be regarded as an extreme form of feature combination as the input embedding pair can be multiplicatively related element by element. Intuitively, different explicit interactions among argument pair can be modeled by each slice of tensor independently.

Trans Model. This model intends to explicitly explore relations between arguments by modeling the relative position information of arguments in the vector space, which can be defined as follows:

$$h = tanh(W_e(a_1 + r - a_2) + W_s[a_1; a_2] + b_s)$$

where $W_e \in \mathbb{R}^{H_2 \times H_1}$ and $r \in \mathbb{R}^{H_1}$.

The transformation operation can be explained as follows: if $Arg1$ and $Arg2$ hold a relation rel, there should be a specific spatial relationship measure that captures the relation between these two arguments. To be straightforward, we expect a transformation embedding r representing relation rel, such that a_1 can correlate to a_2 after adding r. The motivation comes from the work of Mikolov et al. [12], in which the authors state that semantic relations between two words could be found in the embeddings space such as *Paris* - *France* = *Rome* - *Italy*. The work most related to our Trans model is the study of Bordes et al. [2], in which the authors propose a TransE model to learn the entity relations .

TTNN Model. We integrate **T**ensor and **T**rans **N**eural **N**etwork to create a new model called **TTNN** model which is defined as follows:

$$h = tanh({a_1}^T {W_t}^{[1:H_2]} a_2 + W_e(a_1 + r - a_2) + W_s[a_1; a_2] + b_s)$$

Tensor model focuses on the view of interaction between arguments. Trans model intends to explore the view of relative position information of two arguments in the embedding space. Therefore our combined multi-view model should have much more expressive power than each single model.

3.2 Max-margin Learning

After we obtain the relation score of discourse argument pair, we apply max-margin learning framework to optimize the neural network. We define different objective functions for two implicit discourse relation recognition tasks, i.e., binary classification for first-level discourse relations and multiclass classification for second-level discourse relations.

For binary classification, given a training set R of all the (a_1, a_2) pairs with the specific discourse relations, we minimize an objective function defined as follows:

$$L_1(\theta) = \sum_{(a_1,a_2)\in R} \sum_{({a_1}',{a_2}')\notin R} \max\{0, 1 - f(a_1, a_2) + f({a_1}', {a_2}')\} + \lambda \left\|\theta\right\|_2^2$$

For each positive discourse argument pair (a_1, a_2), we randomly sample a certain number of negative pairs $({a_1}^{'}, {a_2}^{'})$ that do not hold the same discourse relation as (a_1, a_2). L_2 regularization is used to penalize the size of all the parameters to prevent overfitting, which is weighted by λ. The objective function L_1 favors higher score for positive training pairs than for negative pairs.

In the testing phase, for each one of the four binary classification sub-tasks, we first use the development set to obtain a threshold T_{rel} for relation rel so that for each argument pair in testing set if $f(a_1, a_2) \geqslant T_{rel}$, then (a_1, a_2) holds the relation rel.

For multiclass classification, we minimize an objective function defined as follows:

$$L_2(\theta) = \sum_{(a_1,a_2)\in R} \sum_{f':f'\neq f} \max\{0, 1 - f^{+}(a_1, a_2) + f^{-}(a_1, a_2)\} + \lambda \left\|\theta\right\|_2^2$$

For each discourse argument pair (a_1, a_2) holding the specific discourse relation rel_i, we score it with $\theta_{rel} = \{W^{rel}, W_s^{rel}, W_t^{rel}, W_e^{rel}, b^{rel}\}_{rel=rel_i}$ as $f^{+}(a_1, a_2)$, and with $\theta_{rel'} = \{W^{rel^{'}}, W_s^{rel^{'}}, W_t^{rel^{'}}, W_e^{rel^{'}}, b^{rel^{'}}\}_{rel'\neq rel_i}$ as $f^{-}(a_1, a_2)$. The objective function L_2 favors higher score for training pairs (a_1, a_2) with series of parameters corresponding to their classes rel_i than with any other series of parameters corresponding to the classes rel' which is not rel_i.

In the testing phase, for each argument pair (a_1, a_2), we score it with the series of parameters for all relations, among which the relation rel with the highest score is held.

4 Multi-level Representations for the Arguments

It is crucial to effectively represent the arguments. Previous work mostly explore various surface features, which cannot capture the features at the segment level. Neural network models can learn segment level information, but the word level information is ignored. Furthermore, syntactic features have been proven to be effective. This motivates us to seek a novel approach that covers not only the multi-level features from token to segments, but also both lexical and syntactic features.

Token Level Lexical Features. Pitler et al. [15] proposes to use the first three and the last words of the argument as features, where connective-like expressions often appear. Thus, we introduce embeddings of tokens that are located at the first three and the last positions of the arguments, which is called **First3-Last** embedding. To obtain the First3-Last embedding, we can simply concatenate or average the embeddings of the first three and the last tokens. The token-level representations can be fine-tuned during training. The token level embeddings are obtained by Word2Vec [1] and we train the model with random initializations.

Segment Level Lexical Features. In addition to token level embeddings, segment embeddings are also indispensable. Segment embeddings learned using a small corpus PDTB[2] through supervised methods [5, 24] cannot beat the surface features, which is in accord with the conclusions of Braud et al. [3]. Paragraph Vector Model, as the augmentation of Word2Vec model, can learn segment level representations in an unsupervised way. In this way, we can obtain the **Sentence embeddings**. Specifically, in order to obtain the **Arguments embeddings**, we assign to them vectors that participate in predicting the target word as sentence vectors do. An example is shown in Figure 1(a). Note that "Sentence" here means an argument pair.

Syntactic Features. To infer the implicit relation between two arguments, the structure difference between them may provide some clues. Lin et al. [8] propose to employ the production rules extracted from constituent parse trees as features and since then these features have been widely used in implicit discourse relation recognition. Some production rule examples in Figure 2 are: S → NP VP, NN → "cat". Li and Nenkova [7] propose a "stick" version of production rules by splitting all the children of a (parent, children) production rule into several sticks where each one only contains one child. For instance, S → NP VP is converted to

[1] https://code.google.com/p/word2vec/.

[2] PDTB contains only 16,053 implicit discourse relation instances

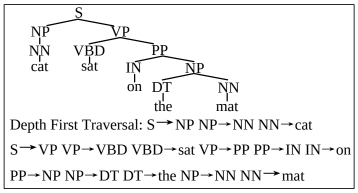

S
NP
VP
NN
VBD
PP
cat
sat
IN
NP
on
DT
NN
the
mat
Depth First Traversal: S→NP NP→NN NN→cat
S→VP VP→VBD VBD→sat VP→PP PP→IN IN→on
PP→NP NP→DT DT→the NP→NN NN→mat

Fig. 2. Linearizations of a parse tree by depth first traversal.

S → NP and S → VP. Through depth-first traversal, we linearize a constituent parser tree to a production stick sequence which is shown in Figure 2. Then, we can learn multi-level distributed representations of syntactic features (in the form of production stick sequences) in the same way as lexical features. The syntactic embeddings are obtained by Word2Vec. For example, the token level syntactic embedding is a representation for each production stick and segment level syntactic embedding is a representation for a production stick sequence.

5 Implementation Details

The optimization is carried out by L-BFGS-B [21] with batch normalization [4] in which we update the model parameters $\{W, W_s, W_t, W_e, b\}$ and word embeddings. We also try AdaGrad but find that it does not work well. We apply norm clipping with a threshold of 5 to overcome the gradient exploding problem and early stop with the development set to avoid overfitting. We select the dimensions of sentences, arguments and word embeddings d among {25, 50, 100}, learning rate η among {0.01, 0.001, 0.0001}, regularization parameter λ among {0.01, 0.001, 0.0001}, number of negative samples for binary classification N among {10, 30, 50, 100}, and size of the hidden layer as well as number of slices in tensor H_2 among {3, 5, 10, 15}. The optimal configurations are determined according to the performance on the development set. The chosen configurations are d=25, η=0.001, λ=0.0001, N=50. For binary classification, H_2=10, while for multiclass classification, H_2=3.

6 Experiments

We test our method on PDTB dataset in two tasks including first-level discourse relation binary classification which attracts more attention recently and second-level discourse relation multiclass classification introduced by Lin et al. [8].

6.1 First-level Relation Recognition

Experimental Settings The task for first-level relation binary classification is to construct a "one-versus-rest" model for each discourse relation. Following the previous work [15, 25, 17] on implicit relation inference, we use sections 2-20 of

PDTB as the training set, sections 0-1 as the development set and sections 21-22 as the test set. Note that the data preparation for Expansion relation follows the work of Zhou et al. [25] and Rutherford and Xue [17]. It is different from the work of Pitler et al. [15] and Ji and Eisenstein [5] in which they regard EntRel relation as a part of Expansion.

To evaluate the effect of syntactic feature on real world data, we do not use the gold standard parse results provided by the Penn Treebank. Our constituent parse results are obtained by using the Stanford Parser [6]. We also employ lowercasing and tokenization. To enlarge the data scale for Paragraph Vector training, we employ a large-scale unlabeled monolingual data from Reuters. From the raw Reuters data, we choose only the sentences in which all the words should appear in PDTB so as to avoid noise. The selected Reuters corpus contains 1.7 billion tokens and 67.2 million sentences. We obtain First3-Last embeddings via an averaging operation and keep them fixed during training. A detailed comparison of different First3-Last embedding compositions will be given on the second-level classification task.

With different lexical and syntactic features, i.e., production rules, we test Single-Layer (SL), Tensor, Trans models and the hybrid TTNN model respectively. The results are reported in Table 2. Finally, we integrate multi-level lexical and syntactic information by summing up the confidence scores obtained by the models with these two features for each instance. Table 3 presents the final performance of our model compared with that of other competitive methods.

Experimental Results In this section, we try to answer three questions: 1) which model for discourse relation scoring performs better; 2) which kinds of distributed features are more effective.

Table 2. The performance (F1-score/%) on recognizing first-level implicit discourse relation with different features and models on PDTB test set.

	SL	Tensor	Trans	TTNN
		Comparison		
Lexical	36.31	40.01	38.29	**41.82**
Syntactic	36.75	39.69	39.34	41.39
		Contingency		
Lexical	48.49	51.30	50.10	52.31
Syntactic	48.53	52.35	51.29	**54.17**
		Expansion		
Lexical	65.69	70.11	71.07	71.03
Syntactic	66.83	70.90	70.91	**71.08**
		Temporal		
Lexical	28.81	31.64	30.09	32.75
Syntactic	29.12	31.81	31.57	**34.04**

The detailed experimental results listed in Table 2 can answer the first two questions. Overall, among all four models, the hybrid TTNN is superior to others, and among the three single models, Tensor model has similar performance to Trans model, which is obviously better than Single-Layer model. Regarding the features, the syntactic features perform better and achieve the best performance in most cases over the four relations.

Table 3. The performance (F1-score/%) for first-level discourse relation classification using multi-level lexical and syntactic features on PDTB test set.

	COM.	CON.	EXP.	TEM.
Pilter et al.(2009)	21.96	47.13	—	16.76
Zhou et al.(2010)	31.79	47.16	65.95	20.30
Rutherford et al.(2014)	39.70	54.42	70.23	28.69
Ji et al.(2015)	35.93	52.78	—	27.63
Braud et al.(2015)	36.36	**55.76**	67.42	29.30
Zhang et al.(2015)	34.22	52.04	69.59	30.54
Liu et al.(2016)	37.91	55.88	69.97	**37.17**
TTNN (ours)	**41.91**	54.72	**71.54**	34.78

The results shown in Table 3 can answer the last question. The experimental results in Table 3 tell us that our method can achieve the best performance in Comparison and Expansion relation recognition tasks when compared to the state-of-the-art approaches (significantly better, McNemar's Chisquared test, p $<$ 0.05). We obtain a competitive result for the Contingency and Temporal relation. These results demonstrate that our method is promising for implicit discourse relation inference.

6.2 Second-level Relation Recognition

Experimental Settings This task belongs to a multiclass classification. Following the work of Ji and Eisenstein [5] in which sections 2-20 of PDTB are used as the training set, sections 0-1 as the development set and sections 21-22 as the test set. There are totally 16 second-level relations while five of them only contains nine samples, and we exclude them as previous work does. For this task, we only implement our TTNN model because it performs best with respect to binary classification. To our knowledge, this is the first work to use the first three and the last one token embeddings to infer discourse relations. Thus, in this task, we conduct extra experiments to evaluate the validity of these token-level embedding features of lexical and syntactic.

We represent First3-Last by concatenating or averaging its token embeddings, and compare First3-Last embedding with sentence and argument embeddings respectively, and then we concatenate First3-Last embedding with sentence and argument embeddings. Moreover, we evaluate whether it is necessary to update the token-level embeddings in training process.

Finally, to compare our model more precisely with other neural network based methods, we choose the best model with distributed representations and add standard surface features as they did. Following Lin et al. [8], we apply feature selection to obtain 500 word pair features, 100 production rule features, 100 dependency rule features and 600 Brown cluster features. The difference lies in the fact that we use information gain (IG) instead of mutual information (MI) as selection criteria because of its better performance[23]. The hidden layer with surface features is defined as follows:

$$h = tanh({a_1}^T {W_t}^{[1:H_2]} a_2 + W_e(a_1 + r - a_2) + W_s[a_1; a_2] + W_{sur} v + b_s)$$

where $W_{sur} \in \mathbb{R}^{H_2 \times d}$ and $v \in \mathbb{R}^d$ is the surface feature vector.

Table 4. The accuracy(%) for second-level discourse relation classification using TTNN with different embeddings on PDTB test set. "Sen", "Arg" and "FL" denote sentence, argument and First3-Last embeddings. "Con." and "ave." denote the concatenating and averaging. "Static" denotes keeping the token embeddings during training while "Dynamic" denotes updating them.

		Lexical	Syntactic
Static	Sen	31.85	32.76
	Arg	37.38	38.09
	Sen+Arg	38.19	38.59
	FL(con.)	32.76	29.44
	FL(ave.)	34.57	31.15
	Sen+Arg+FL(con.)	38.09	39.20
	Sen+Arg+FL(ave.)	**40.90**	39.89
Dynamic	Sen+Arg+FL(con.)	35.97	36.20
	Sen+Arg+FL(ave.)	40.70	39.49

Experimental Results Table 4 can answer three questions about the embedding layer: 1) what type of embeddings are more effective; 2) what type of First3-Last representation is better; and 3) do we need to update the token embeddings. As shown in the first five lines of Table 4, argument embeddings are the most effective while sentence embeddings are the worst. From the remaining lines in Table 4, we can conclude that averaging is better than concatenation for First3-Last embedding composition. Although concatenation can introduce the word order information, it may lead to the sparsity problem due to separate treatments being used for each word located at the first three and last position of the arguments. Updating the token-level embeddings does not contribute to the classification accuracy perhaps because of the overfitting problem in this model.

Table 5. Performance (Accuracy/%) comparison for second-level implicit discourse relation classification on PDTB test set.

Models		Accuracy
Surface features based models	Lin et al., (2009)	40.20
	Ji & Eisenstein, (2015)	**40.66**
	TTNN (ours)	40.52
Neural networks based models	Ji & Eisenstein, (2015)	36.98
	Rutherford et al., (2016)	39.56
	TTNN with lexical features	40.90
	TTNN with syntactical features	39.94
	TTNN with lexical and syntactical features	**41.39**
Neural networks based models + surface features	Ji & Eisenstein, (2015)	44.59
	TTNN	**44.75**

Table 5 shows the final results of our model compared with other competitive systems. For surface features, our model achieves the performance similar to that of the other two systems. When excluding surface features, the classification accuracy of our model is the best, with a 4.41% and 1.83% improvement over the system of Ji and Eisenstein (2015) and Rutherford et al. (2016). Note that their models do not beat Lin's purely surface feature model. In contrast, our model outperforms the surface features based model (statistically significant, p < 0.01; t-test). Finally, when surface features are further added to our model, we can achieve the best accuracy of 44.75%. Note that the bilinear model of Ji and Eisenstein (2015) can be regarded as a special case of the one-way Tensor model without a hidden layer and the transformation property is out of their consideration, so that our model has much more expressive power.

7 Discussion

To better understand the strength of our multi-level distributed representations, some discourse relation instances which are extracted from the test set of PDTB are given in Table 1. The recognition results of these instances are incorrect by the model using discrete surface features while correct using the distributed representations. We first explain the necessity of our distributed First3-Last embeddings and then we explore the deeper reasons at the sentence level.

Pitler et al. [15] proposes that connective-like expressions appear at the first three and the last words of the arguments and we find that our distributed First3-Last embeddings have advantage over their discrete features. We demonstrate this using the first example in Table 1. In Example (1) of Table 1, the first three words of $Arg1$ - "The common view" and the first two words of $Arg2$ - "Our view" indicate that it will pose opposite opinions for the two arguments, thus, the Contrast relation exists between the argument pair. However, this rule does not appear in the training set of PDTB. In other words, it is impossible to detect the discourse relation by using the discrete First3-Last features. In contrast, our distributed First3-Last representation can capture these connective-like expressions and recognize the discourse relation successfully.

Next, we show the effectiveness of segment-level distributed representations. Lin et al. [8] explains that implicit discourse relation recognition needs a deeper semantic representation and a more robust model. Distributed representation of word or segment and neural network model may meet these requirements. Regarding example (2) in Table 1, $Arg2$ is an instantiation of $Arg1$. Word pair "declined, off" provides a strong indication for this discourse relation, but we find that such a case does not occur in the training set, thus it is not surprising that using surface features including word pairs fails to detect the relation.

For our distributed representation based model, the recognition result is correct. We seek the most similar argument in the PDTB training set of $Arg1$ and $Arg2$ using cosine similarity in vector space, yielding "For the first nine months, the trade deficit was 14.933 trillion lire, compared with 10.485 trillion lire in the year-earlier period" (denoted as $Arg1^{'}$) for $Arg1$ and "The stock fell 75 cents" (denoted as $Arg2^{'}$) for $Arg2$. We find that there is a few words overlap between $Arg1^{'}$ and $Arg1$ as well as between $Arg2^{'}$ and $Arg2$, but there is a relatively high semantic similarity between of $Arg1^{'}$ and $Arg1$, and also between $Arg2^{'}$ and $Arg2$: $Arg1^{'}$ and $Arg1$ both express the meaning of slowdown in terms of the economy; $Arg2^{'}$ and $Arg2$ express that the price of something decreases by a specific number of cents.

According to the analyses above, we can conclude that our model has the ability to capture deeper semantic meaning while the discrete surface feature model fails.

8 Related Work

Most of the previous work [15, 8, 25, 22] regards implicit discourse relation recognition as a classification task that focuses on feature engineering. Subsequent work [1, 7, 19, 17] focuses on addressing the data sparsity problem. Recently, deep learning methods [5, 24, 9, 18] have been applied to this task. Ji and Eisenstein [5] employs a recursive neural network and achieves state-of-the-art performance for second-level relations. However, without the surface features, the performance of Ji and Eisenstein [5] model is about 3% lower than surface features based model of Lin et al. [8]. Part of the reason may be that it is difficult to learn satisfying representations of sentence with small-sized PDTB corpus.

9 Conclusion

In this paper, we proposed a novel method for implicit discourse relation recognition based on neural network in which the model hierarchy and the feature hierarchy are proposed. Regarding the model hierarchy, we propose a max-margin neural network that considers two views, including the relation classification view and the relation transformation view. Regarding the feature hierarchy, we learn and leverage distributed representations from multi-levels, namely from words, arguments and syntactic structures to sentences.

We test our method in implicit discourse relation binary classification and multi-class prediction. The experimental results demonstrate that our method can achieve new state-of-the-art performance in most cases. Furthermore, we find for the first time that the distributed features can perform better than surface discrete features for second-level implicit discourse relation recognition.

Acknowledgments

The research work has been funded by the Natural Science Foundation of China under Grant No. 61403379.

References

1. Biran, O., McKeown, K.: Aggregated word pair features for implicit discourse relation disambiguation. In: Proceedings of the Conference. p. 69 (2013)
2. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems. pp. 2787–2795 (2013)
3. Braud, C., Denis, P.: Comparing word representations for implicit discourse relation classification. In: EMNLP 2015 (2015)
4. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
5. Ji, Y., Eisenstein, J.: One vector is not enough: Entity-augmented distributed semantics for discourse relations. Transactions of the Association of Computational Linguistics – Volume 3, Issue 1 (2015), http://aclweb.org/anthology/Q15-1024

6. Klein, D., Manning, C.D.: Accurate unlexicalized parsing. In: Proceedings of A-CL2003. pp. 423–430 (2003)
7. Li, J.J., Nenkova, A.: Reducing sparsity improves the recognition of implicit discourse relations. In: 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue. p. 199 (2014)
8. Lin, Z., Kan, M.Y., Ng, H.T.: Recognizing implicit discourse relations in the penn discourse treebank. In: Proceedings of EMNLP2009 (2009)
9. Liu, Y., Li, S., Zhang, X., Sui, Z.: Implicit discourse relation classification via multi-task neural networks. arXiv preprint arXiv:1603.02776 (2016)
10. Louis, A., Joshi, A., Prasad, R., Nenkova, A.: Using entity features to classify implicit discourse relations. In: Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue. pp. 59–62 (2010)
11. Marcu, D., Echihabi, A.: An unsupervised approach to recognizing discourse relations. In: Proceedings of ACL 2002. pp. 368–375. Association for Computational Linguistics (2002)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Park, J., Cardie, C.: Improving implicit discourse relation recognition through feature set optimization. In: Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (2012)
14. Pei, W., Ge, T., Baobao, C.: Maxmargin tensor neural network for chinese word segmentation. In: Proceedings of ACL (2014)
15. Pitler, E., Louis, A., Nenkova, A.: Automatic sense prediction for implicit discourse relations in text. In: Proceedings of ACL2009. Proceedings of ACL 2009 (2009)
16. Prasad, R., Lee, A., Miltsakaki, E., Robaldo, L., Joshi, A.K., Webber, B.L., Dinesh, N.: The penn discourse treebank 2.0. In: Lrec 2008. pp. 2961–2968 (2008)
17. Rutherford, A., Xue, N.: Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In: Proceedings of NAACL2015. pp. 799–808. Association for Computational Linguistics (2015), `http://aclweb.org/anthology/N15-1081`
18. Rutherford, A.T., Demberg, V., Xue, N.: Neural network models for implicit discourse relation classification in english and chinese without surface features (2016)
19. Rutherford, A.T., Xue, N.: Discovering implicit discourse relations through brown cluster pair representation and coreference patterns. EACL 2014 p. 645 (2014)
20. Socher, R., Chen, D., Manning, C.D., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. In: Advances in Neural Information Processing Systems. pp. 926–934 (2013)
21. Tang, W., Zhang, L., Linninger, A.A., Tranter, R.S., Brezinsky, K.: Solving kinetic inversion problems via a physically bounded gauss-newton (pgn) method. Industrial & engineering chemistry research 44(10), 3626–3637 (2005)
22. Xu, Y., Lan, M., Lu, Y., Niu, Z.Y., Tan, C.L.: Connective prediction using machine learning for implicit discourse relation classification. In: Neural Networks (IJCNN), The 2012 International Joint Conference on. pp. 1–8. IEEE (2012)
23. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: ICML. vol. 97, pp. 412–420 (1997)
24. Zhang, B., Su, J., Xiong, D., Lu, Y., Duan, H., Yao, J.: Shallow convolutional neural network for implicit discourse relation recognition. In: Proceedings of EMNLP2015
25. Zhou, Z.M., Xu, Y., Niu, Z.Y., Lan, M., Su, J., Tan, C.L.: Predicting discourse connectives for implicit discourse relation recognition. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. pp. 1507–1514. Association for Computational Linguistics (2010)