Multi-labeled Relation Extraction with Attentive Capsule Network

Xinsong Zhang[1], **Pengshuai Li**[1], **Weijia Jia**[2,1*], **and Hai Zhao**[1*]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2]State Key Lab of IoT for Smart City, University of Macau, Macau 999078, China
{xszhang0320, pengshuai.li}@sjtu.edu.cn and {jia-wj, zhaohai}@cs.sjtu.edu.cn

Abstract

To disclose overlapped multiple relations from a sentence still keeps challenging. Most current works in terms of neural models inconveniently assuming that each sentence is explicitly mapped to a relation label, cannot handle multiple relations properly as the overlapped features of the relations are either ignored or very difficult to identify. To tackle with the new issue, we propose a novel approach for multi-labeled relation extraction with capsule network which acts considerably better than current convolutional or recurrent net in identifying the highly overlapped relations within an individual sentence. To better cluster the features and precisely extract the relations, we further devise attention-based routing algorithm and sliding-margin loss function, and embed them into our capsule network. The experimental results show that the proposed approach can indeed extract the highly overlapped features and achieve significant performance improvement for relation extraction comparing to the state-of-the-art works.

Introduction

Relation extraction plays a crucial role in many natural language processing (NLP) tasks. It aims to identify relation facts for pairs of entities in a sentence to construct triples like [*Arthur Lee, place_born, Memphis*]. Relation extraction has received renewed interest in the *neural network* era, when neural models are effective to extract semantic meanings of relations. Compared with traditional approaches which focus on manually designed features, neural methods such as Convolutional Neural Network (CNN) (Liu et al. 2013; Zeng et al. 2014) and Recurrent Neural Network (RNN) (Zhang and Wang 2015; Zhou et al. 2016) have achieved significant improvement in relation classification. However, previous neural models are unlikely to scale in the scenario where a sentence has multiple relation labels and face the challenges in extracting highly overlapped and discrete relation features due to the following two drawbacks.

First, one entity pair can express multiple relations in a sentence, which will confuse relation extractor seriously. For example, as in Figure 1, the entity pair [*Arthur Lee, Memphis*] keeps three possible relations which are *place_birth*, *place_death* and *place_lived*. The sentence *S1* and *S2* can both express two relations, and the sentence *S3* represents another two relations. These sentences contain multiple kinds of relation features which are difficult to be identified clearly. The existing neural models tendentiously merge low-level semantic meanings to one high-level relation representation vector with methods such as max-pooling (Zeng et al. 2014; Zhang, Zhao, and Qin 2016) and word-level attention (Zhou et al. 2016). However, one high-level relation vector is still insufficient to express multiple relations precisely.

Second, current methods are neglecting of the discretization of relation features. For instance, as shown in Figure 1, all the sentences express their relations with a few significant words (labeled italic in the figure) distributed discretely in the sentences. However, common neural methods handle sentences with fixed structures, which are difficult to gather relation features of different positions. For example, being spatially sensitive, CNNs adopt convolutional feature detectors to extract local patterns from a sliding window of vector sequences and use the max-pooling to select the prominent ones. Besides, the feature distribution of "no relation (NA, others)" in a dataset is different from that of definite relations. A sentence can be classified to "no relation" only when it does not contain any features of other relations.

In this paper, to extract overlapped and discrete relation features, we propose a novel approach for multi-labeled relation extraction with an attentive capsule network. As shown in Figure 1, the relation extractor of the proposed method is constructed with three major layers that are feature extracting, feature clustering and relation predicting. The first one extracts low-level semantic meanings. The second layer clusters low-level features to high-level relation representations, and the final one predicts relation types for each relation representation. The low-level features are extracted with traditional neural models such as Bidirectional Long Short-Term Memory (Bi-LSTM) and CNN. For the feature clustering layer, we utilize an attentive capsule network inspired by Sabour, Frosst, and Hinton (2017). Capsule (vector) is a small group of neurons used to express features. Its overall length indicates the significance of features, and the direction of a capsule suggests the specific property of the feature. The low-level semantic meanings from the first layer are embedded to amounts of low-level capsules, which will

*Corresponding authors: Weijia Jia, Hai Zhao, {jia-wj, zhaohai}@cs.sjtu.edu.cn

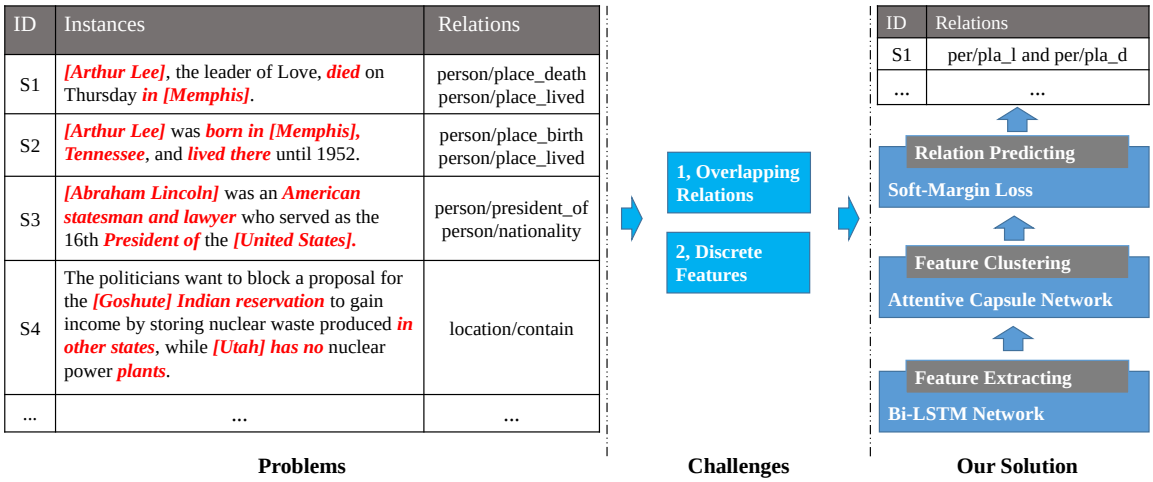

Figure 1: Problems, challenges and our solution for multi-labeled relation extraction. Words in brackets are entities and the italic red parts are key words that contain relation features. (The relation label in the right table is in abbreviation.)

be routed and clustered together to represent high-level relation features. For better relation extraction, we further devise an attention-based routing algorithm to precisely find low-level capsules that contain related relation features. Besides, we propose a sliding-margin loss function to address the problem of "no relation" in multiple labels scenario. A sentence is classified as "no relation" only when the probabilities for all the other specific classes are below a boundary. The boundary is dynamically adjusted in the training process. Experimental results on two widely used benchmarks show that the proposed method can significantly enhance the performance of relation extraction. The contributions of this paper can be summarized as follows,

- We first apply capsule network to multi-labeled relation extraction by clustering relation features.
- We propose an attention-based routing algorithm to precisely extract relation features and a sliding-margin loss function to well learn multiple relations.
- Our experiments on two benchmarks show our method gives new state-of-the-art performance.

Related work

Relation Extraction. Relation extraction is a critical task for the NLP in which supervised methods with human-designed features have been well studied (Mooney and Bunescu 2006; Mintz et al. 2009; Riedel, Yao, and McCallum 2010; Surdeanu et al. 2012). Recent years, neural models are widely used to remove the inconvenience of hand-crafted feature design. Both CNN and RNN have been well applied to relation extraction (Socher et al. 2012; Liu et al. 2013; Kim 2014; Zeng et al. 2014; Santos, Xiang, and Zhou 2015; Zhang and Wang 2015). From the CNN or RNN backbone, relation extraction can be further improved by integrating attention mechanism (Wang et al. 2016; Lin et al. 2016; Zhou et al. 2016; Zhu et al. 2017), parser tree (Xu et al. 2015; Miwa and Bansal 2016; Xu et al. 2016), multi-task learning (Liu, Qiu, and Huang 2016) or ensemble models (Nguyen and Grishman 2015; Yang, Wang, and Li 2018). However, all the previous neural models simply represent relation features with one vector, resulting in unacceptable precisions for multi-labeled relation extraction.

Capsule Network. Capsule network was proposed to improve the representational limitations of CNN and RNN (Hinton, Krizhevsky, and Wang 2011). Capsules with transformation matrices allow networks to learn part-whole relationships automatically. Consequently, a dynamic routing algorithm (Sabour, Frosst, and Hinton 2017) was proposed to replace the max-pooling in CNN, which achieved impressive performance recognizing highly overlapping digits. Then, Xi, Bing, and Jin (2017) further tested out the application of capsule networks on the CIFAR data with higher dimensionality. Hinton, Sabour, and Frosst (2018) proposed a new routing method between capsule layers based on the EM algorithm. Recently, capsule network was applied to NLP tasks such as text classification (Zhao et al. 2018) and disease classification (Wang et al. 2018).

Different from the previous methods for multi-labeled relation extraction, we first introduce the capsule network to the task, especially, with two highlighted improvements, attentive routing algorithm and sliding-margin loss.

Method

This section describes our approach for multi-labeled relation extraction with an attentive capsule network. As shown in Figure 2, our relation extractor comprises of three primary layers,

- **Feature Extraction Layer.** Given a sentence b^* and two target entities, a Bi-LSTM network is used to extract low-level features of the sentence.
- **Feature Clustering Layer.** Given vectors of low-level features, we cluster the related features into a high-level

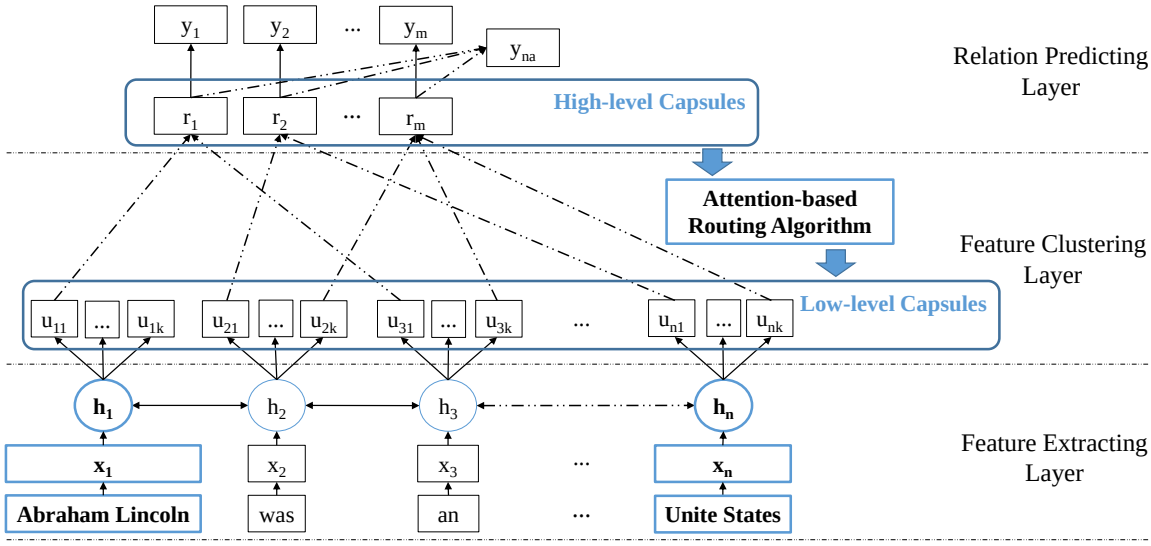

Figure 2: The architecture of our proposed relation extractor, illustrating the procedure for handling one sentence and predicting possible relations between [*Abraham Lincoln*] and [*Unite States*]. h is a set of hidden states of Bi-LSTM, u is a low-level capsule set and r represents high-level capsules. y indicates relation labels, and y_{na} expresses "no-relation". The solid lines are determinate associations, and the dotted lines are possible ones.

relation representation for each relation with an attentive capsule network.

- **Relation Predicting Layer.** After computing the high-level representations of a sentence, we apply a sliding-margin loss function to predict possible relations including the label "no relation".

Feature Extracting Layer

This layer encodes word tokens and extracts low-level semantic information.

Input Representation The input representations of our model include word embeddings and position embeddings.
Word Embeddings are distributed representations of words that map each word to a p dimensional real-valued vector $word$. The vectors are pre-trained in the $skip-gram$ setting of $word2vec$ (Mikolov et al. 2013).
Position Embeddings are defined as the combination of the relative distances from the current word to the entities. For instance, in sentence *Arthur Lee was born in Memphis.*, the relative distances from the word *born* to [*Arthur Lee*] and [*Memphis*] are respectively 2 and -2. We also encode distances to vectors $position \in \mathcal{R}^q$, where q is the dimension of position embeddings. In our work, the position embeddings are initialized randomly.

The word embeddings and position embeddings are concatenated together as network input vector. We denote all the words in a sentence as an initial vector sequence $b^* = \{x_1, \cdots, x_i, \cdots, x_n\}$, where $x_i \in \mathcal{R}^{p+q}$ and n is the word number.

Bidirectional LSTM Recurrent Neural Network (RNN) is powerful to model sequential data and has achieved great success in the relation classification (Zhang and Wang 2015; Zhou et al. 2016). RNN for our task is implemented as LSTM with four components (Graves 2013), one input gate i_t with corresponding weight matrix W_i, U_i, V_i, one forget gate f_t with corresponding weight matrix W_f, U_f, V_f, one output gate o_t with corresponding weight matrix W_o, U_o, V_o and one cell c_t with corresponding weight matrix W_c, U_c, V_c. All of those components are set to generate the current hidden state h_t with the input token x_t and the previous hidden state h_{t-1}. The whole procedure is demonstrated with the following equations,

$$
\begin{aligned}
i_t &= \sigma(W_i[x_t] + U_i h_{t-1} + V_i c_{t-1} + b_i) \\
f_t &= \sigma(W_f[x_t] + U_f h_{t-1} + V_f c_{t-1} + b_f) \\
c_t &= i_t \tanh(W_c[x_t] + U_c h_{t-1} + V_c c_{t-1} + b_c) + f_t c_{t-1} \\
o_t &= \sigma(W_o[x_t] + U_o h_{t-1} + V_o c_t + b_o) \\
h_t &= o_t \tanh(c_t),
\end{aligned}
$$

where σ is the sigmoid function and all of the components have the same sizes as the hidden vector h.

For many sequence modeling tasks, it is beneficial to have access to the future context as well as the past. Bidirectional-LSTM network extends the standard LSTM network by introducing a second layer, in which the hidden states flow in an opposite temporal order. The model is, therefore, able to exploit features both from the past and the future. Consequently, we use Bi-LSTM including both forward subnetwork and backward subnetwork to capture global sequence information. The final state of h_t is shown by the equation $h_t = [\overrightarrow{h_t} \oplus \overleftarrow{h_t}]$, where $\overrightarrow{h_t}$ is the forward state, $\overleftarrow{h_t}$ is the backward state and $\oplus$ is the element-wise sum. The dimension of the hidden state is determined by a hyper-parameter s_h.

Feature Clustering Layer

This layer clusters features with the help of an attention-based routing algorithm.

Feature Clustering with Capsule Network Capsule network has been proved effective in digital recognition, especially for the highly overlapping digits (Sabour, Frosst, and Hinton 2017). In our work, a capsule is a group of neurons whose activity vector represents the instantiation parameters of a specific type of relation features. The length of the activity vector represents the probability that the relation features exist, and the orientation of the vector expresses the specific property of one kind of features. Active capsules make predictions, via transformation matrices, for the instantiation parameters of higher-level capsules. While, high-level capsules are clustered from low-level capsules, which contain local and trivial features. When multiple predictions agree, a higher level capsule becomes active.

For the task of relation extraction, we scatter all the low-level semantic information extracted by the Bi-LSTM into amounts of low-level capsules represented by $u \in \mathcal{R}^{d_u}$. The representation of each word token will be expressed by k low-level capsules. Each low-level capsule is applied with a nonlinear squash function g through the entire vector,

$$h_t = [u^{'}_{t1}; \cdots ; u^{'}_{tk}]$$
$$u_{tk} = g(u^{'}_{tk}) = \frac{||u^{'}_{tk}||^2}{1 + ||u^{'}_{tk}||^2} \frac{u^{'}_{tk}}{||u^{'}_{tk}||},$$

where $[x; y]$ denotes the vertical concatenation of x and y. Amounts of low-level capsules can be clustered together to represent high-level relation features. Therefore, high-level capsules $r \in \mathcal{R}^{d_r}$ are computed with the following equations,

$$r_j = g(\sum_i w_{ij} W_j u_i),$$

where w_{ij} are coupling coefficients that are determined by an iterative dynamic routing process and $W_j \in \mathcal{R}^{d_r \times d_u}$ are weight matrices for each high-level capsule.

Attention-based Routing Algorithm With the capsule network, we can obtain high-level capsules which represent relation features. However, the traditional dynamic routing algorithm in (Sabour, Frosst, and Hinton 2017) does not focus on the entity tokens, which have been proved important for relation extraction (Wang et al. 2016; Liu et al. 2018). Therefore, we propose an attention-based routing algorithm which focuses on the entity tokens when routing for related low-level capsules as in Algorithm 1. The coupling coefficients w between i-th capsule and all the capsules in the r sum to 1 and are determined by a "softmax" function whose initial logits are b_{ij}, the log prior probabilities that capsule u_i should be coupled to capsule r_j. Besides, we propose attention weights α for all the low-level capsules to maximize the weights of capsules from significant word tokens and minimize that of irrelevant capsules. The weight of the capsule u_i is computed by the entity features h_e and hidden state h^i_t from which the u_i comes. The w and α are computed by

$$w_{ij} = \frac{\exp(b_{ij})}{\sum_{j^*} \exp(b_{ij^*})}$$
$$\alpha_i = \sigma(h_e^T h_t^i),$$

where h_e is the sum of hidden states of the two entities and T means the transpose operation. The sigmoid function normalizes the attention weights and maximizes the differences between significant capsules and irrelevant ones. Finally, the high-level capsules r are computed with the Algorithm 1.

Algorithm 1 Attention-based Routing Algorithm

Require: low-level capsules u, iterative number z, entity features h_e and hidden states h_t
Ensure: high-level capsules r
for all capsules u_i and capsules r_j **do**
  initialize the logits of coupling coefficients
  b_{ij} = 0
end for
for z iterations **do**
  w_i = softmax(b_i), $\forall u_i \in u$
  $\alpha_i = \sigma(h_e^T h_t^i)$, $\forall u_i \in u$
  $r_j = g(\sum_i w_{ij} \alpha_i W_j u_i)$, $\forall r_j \in r$
  $b_{ij} = b_{ij} + W_j u_i r_j$, $\forall u_i \in u$ and $\forall r_j \in r$
end for

Relation Predicting Layer

In the capsule network, the length of the activity high-level capsules can represent the probability of relations. Sabour, Frosst, and Hinton (2017) applied a fixed margin loss for the classification of digit images. However, they can only set the margin empirically. Therefore, we propose a sliding-margin loss for the task of relation extraction, which learns the baseline of the margin automatically. Besides, to deal with the "no relation" (presented as NA in the figure 2) properly, the probabilities of all the existing relations for sentences labeled as NA should be under the lower bound of the margin. The loss function for the j-th relation follows the below equation,

$$L_j = Y_j max(0, (B + \gamma) - ||r_j||)^2 +$$
$$\lambda(1 - Y_j) max(0, ||r_j|| - (B - \gamma))^2,$$

where $Y_j = 1$ if the sentence represents relation r_j, and $Y_j = 0$ if not. γ is a hyper-parameter defining the width of the margin, and B is a learnable variable indicating the NA threshold of the margin, which is initialized by 0.5. λ is the down-weighting of the loss for absent relations, which is the same as that in (Sabour, Frosst, and Hinton 2017). The total loss of a sentence is the sum of losses from all the relations. In the testing process, relation labels will be assigned to a sentence when its probabilities of these relations are larger than the threshold B. Otherwise, it will be predicted as NA.

Experiments

We conduct experiments to answer the following three questions. 1) Does our method outperform previous works in relation extraction? 2) Is attentive capsule network useful to

distinguish highly overlapping relations? 3) Are the two proposed improvements both effective for relation extraction?

Dataset, Evaluation Metric and Baselines

Dataset. We conduct experiments on two widely used benchmarks for relation extraction, NYT-10 (Riedel, Yao, and McCallum 2010) and SemEval-2010 Task 8 dataset (Hendrickx et al. 2009). The NYT-10 dataset is generated by aligning Freebase relations with the New York Times (NYT) corpus, in which sentences from the years 2005-2006 are for training while those from 2017 for testing. The dataset consists of amounts of multi-labeled sentences. The SemEval-2010 Task 8 dataset is a small dataset which has been well-labeled for relation extraction. The details of both datasets are shown in Table 1.

Datasets	Train Sen.	Test Sen.	Multi-labeled Sen.	Classes
NYT-10	566,190	170,866	45,693	53
Sem.	8,000	2,717	0	19

Table 1: Detail information for datasets. **Sen.** is the number of sentences. **Sem.** represents SemEval-2010 Task 8.

Evaluation Metric. We evaluate our method with a classical held-out evaluation for NYT-10 and macro-averaged F1 for SemEval-2010 Task 8. The held-out evaluation evaluates our models by comparing the relation facts discovered from the test articles with those in Freebase, which provides an approximation of the precision without the time-consuming human evaluation. Besides, we report both the aggregate Precision-Recall (PR) curves and macro-averaged F1 as quantitative indicators.

Baselines. We select following feature clustering methods as baselines.

Max-pooling+CNN clusters the relations extracted by CNN with max-pooling (Zeng et al. 2014).

Max-pooling+RNN clusters the relations extracted by RNN with max-pooling (Zhang and Wang 2015).

Avg+RNN aggregates the relation features with linear average of all the hidden states of word tokens.

Att+RNN applies a word-level attention to aggregate relation features instead of linear average (Zhou et al. 2016).

Att-CapNet (CNN-based) integrates our attentive capsule network with a CNN relation extractor.

Att-CapNet (RNN-based) is our method.

Experimental Settings

In our experiments, word embeddings are pre-trained with the $word2vec$ tool. For NYT-10, we pre-train the word embeddings on NYT-10 in the $skip-gram$ setting. In order to compare with the previous works on SemEval-2010 Task 8, we use the same word vectors proposed by (Turian, Ratinov, and Bengio 2010) (50-dimensional) to initialize the embedding layer. Additionally, we also use the 100-dimensional word vectors pre-trained in $Glove$ setting (Pennington, Socher, and Manning 2014). Besides, we concatenate the words of an entity when it has multiple words. Position embeddings are initialized randomly and updated in training. We use Adam optimizer (Kingma and Ba 2015) to minimize the objective function. $L2$ regularization and dropout (Srivastava et al. 2014) are adopted to avoid overfitting. To train our model efficiently, we iterate by randomly selecting a batch from the training set until convergence. We use a grid search to determine the optional parameters. Table 2 lists our hyper-parameter setting[1].

Parameters	NYT-10	Sem.
batch size	50	50
word dimension p	50	50
position dimension q	5	5
hidden state dimension s_h	256	256
capsule dimensions $[d_u, d_r]$	[16,16]	[16,16]
iterations z	3	3
sliding-margin γ	0.4	0.4
down-weighting λ	1.0	0.5
learning rate	0.001	0.001
dropout probability	0.0	0.7
L2 regularization strength	0.0001	0.0

Table 2: Parameter settings

Overall Performance

We compare our method with the previous baselines on the two datasets respectively. For the dataset NYT-10, the performance of all the methods is compared in Figure 3. The figure draws the PR curves of all the baselines. Apparently, we can see, 1) our Att-CapNet (RNN-based) model achieves the best PR curve, which outperforms the other baselines at nearly all range of the recall. 2) our Att-CapNet (CNN-based) model is slightly better than the method Max-pooling+CNN (Zeng et al. 2014). 3) RNN models tendentiously outperform CNN ones for their strong ability of extracting low-level relation features from the sequence. 4) our attentive capsule network is effective for relation extraction integrated with either CNN or RNN.

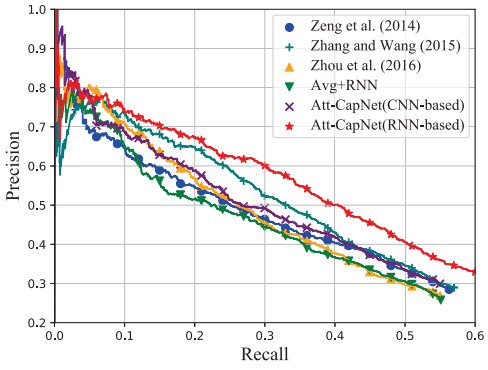

Figure 3: The PR curves of all the baselines on NYT-10

[1]The parameters of the baselines are following their papers.

A detailed comparison of baselines with precision, recall, F1 and PR curve areas is shown in Table 3, which indicates, 1) our Att-CapNet (RNN-based) obtains better results on all the indicators than the baselines and increases F1 scores by at least 3.2% over the other baselines. 2) Att-CapNet (CNN-based) is slightly better than Max-pooling+CNN (Zeng et al. 2014). 3) the previous feature extractors such as RNN and CNN are all improved by integrating with our attentive capsule network.

Methods	Precision(%)	Recall(%)	F1(%)	PR
Zeng et al. (2014)	28.5	56.3	37.8	0.35
Zhang and Wang (2015)	28.9	57.0	38.4	0.34
Zhou et al. (2016)	26.9	54.9	36.1	0.34
Avg+RNN	25.7	55.1	35.1	0.33
Att-CapNet (CNN-based)	29.9	55.0	38.8	0.36
Att-CapNet (RNN-based)	**30.8**	**63.7**	**41.6**	**0.42**

Table 3: Performance of all the baselines on NYT-10. PR represents precision-recall curve area.

We further conduct paired t-test (10-fold, F1 score) to evaluate the statistical significance of our results in terms of p-value and confidence intervals. Table 4 shows that all the p-values are less than 5.0e-02 and the increases in F1 score are at least 2.2%. Therefore, all our performance improvements are statistically significant.

Baselines	p-value	CI (Confident level 95%)
Zeng et al. (2014)	1.0e-02	[0.023, 0.051]
Zhang and Wang (2015)	1.2e-02	[0.022, 0.041]
Zhou et al. (2016)	2.4e-05	[0.052, 0.077]
Avg+RNN	1.1e-04	[0.044, 0.064]

Table 4: The statistical significance in the difference between Att-CapNet (RNN-based) and the baselines. CI represents confidence intervals.

From the results in Table 5 on SemEval-2010 dataset[2], we can have the following observations, 1) our Att-CapNet (RNN-based) outperforms all the other feature clustering methods. 2) Att-CapNet (CNN-based) is better than the traditional CNN model with max-pooling. 3) RNN models are better than CNN ones under the same settings. 4) our attentive capsule network is more useful than the other feature clustering methods such as max-pooling or word-level attention.

Effect of Our Method on Multi-labeled Sentences

To evaluate the effect of our method on the multi-labeled sentences, we randomly select 500 sentences, which have more than one labels, from NYT-10 for testing. The previous methods cannot predict multiple relations, and all of them can only obtain a low recall rate of about 0.40. Therefore, we define a threshold confident score for all the previous methods to make them predict multiple relations. We tune the threshold to ensure that all the previous methods can achieve maximum F1 scores[3]. As shown in Table 6, our Att-CapNet (RNN-based) achieves the best precision, recall and F1. Our methods can recall more relation labels in the scenario where a sentence contains different relations.

Methods	Features	F1(%)
Zeng et al. (2014)	WE (dim=50)	69.7
Zeng et al. (2014)‡	WE (dim=50)+PE	79.8
Zhang and Wang (2015)	WE (dim=50)	80.0
Zhang and Wang (2015)	WE (dim=300)	82.5
Zhang and Wang (2015)‡	WE (dim=50)+PE	81.0
Zhou et al. (2016)	WE (dim=50)	82.5
Zhou et al. (2016)	WE (dim=100)	84.0
Zhou et al. (2016)‡	WE (dim=50)+PE	81.7
Avg+RNN‡	WE (dim=50)+PE	78.4
Att-CapNet (CNN-based)	WE (dim=50)+PE	80.4
Att-CapNet (RNN-based)	WE (dim=50)+PE	**84.5**

Table 5: Performance of all the baselines on SemEval-2010 Task 8. WE, PE respectively stand for word embedding and position embedding. Methods with ‡ are our implementations. The other results are reported in their papers.

Methods	Precision(%)	Recall(%)	F1(%)
Max-pooling+CNN	88.4	91.9	90.1
Max-pooling+RNN	89.3	91.8	90.5
Att+RNN	88.8	90.6	89.7
Avg+RNN	86.9	90.5	88.6
Att-CapNet (CNN-based)	87.3	93.0	90.1
Att-CapNet (RNN-based)	**89.9**	**93.7**	**91.8**

Table 6: Performance of all the baselines on selected 500 multi-labeled sentences from NYT-10.

Effect of Various Modules

In this subsection, we evaluate various modules of our method including attention-based routing algorithm and sliding-margin loss function. Our main method outperforms the other variants, although the variants may still prove useful when applied to other tasks. We apply our model Att-CapNet (RNN-based) and its two sub-models, which are without the attention-based routing algorithm (dynamic routing) and sliding-margin loss (fixed-margin loss), to the two datasets respectively. The results shown in Table 8 and Figure 4 indicate, 1) our main model and the two variants are better than the best baseline feature clustering method under the same settings. 2) our attention-based routing algorithm and sliding-margin loss are both useful for capsule network, which significantly enhance the performance of relation extraction.

[2]We compare our method with previous works which do not depend on the parser information and external data such as WordNet.

[3]We compute F1 scores for a series of confidence scores and select the maximum ones for the previous methods. The interval of confidence scores is 0.1.

Sentences	Labels	RNN			
		Max-pooling	Avg.	Att.	Att-CapNet
S1: Twenty years ago, another **[Augusta]** native, **[Larry Mize]**, shocked Greg Norman in a playoff by holing a 140-foot chip for birdie on the 11th hole to win the masters in a playoff.	PB	0	0	0	1
	PL	0	1	1	1
S2: Brothers or cousins except for its drummer, Oscar Lara, the band originally comes from **[Sinaloa]**, **[Mexico]**, but has lived in San Jose, California, for nearly 40 years.	LC	1	0	0	1
	CA	0	1	1	1
S3: The white house in April sharply criticized the speaker of the house, Nancy Pelosi, for visiting **[Syria]**'s capital, **[Damascus]**, and meeting with president Bashar Al-Assad, even going so far as calling the trip "bad behavior", in the words of vice president Dick Cheney.	LC	0	0	0	1
	CA	0	0	1	1
	CC	1	1	0	1

Table 7: A case study of selective multi-labeled sentences for the four feature clustering methods based on RNN. The entities are labeled in the bold brackets. "PB", "PL", "LC", "CA" and "CC" are relation labels in the dataset, which are "person/place_birth", "person/place_lived","location/contain", "country/administrative_divisions" and "country/capital" respectively.

Methods	Features	F1(%)
Zhou et al. (2016)	WE (dim=50)+PE	81.7
Att-CapNet (RNN-based)	WE (dim=50)+PE	**84.5**
-w/o attention-based routing	WE (dim=50)+PE	83.6
-w/o sliding-margin loss	WE (dim=50)+PE	82.3

Table 8: Performance of Att-CapNet (RNN-based) with various modules on SemEval-2010 Task 8.

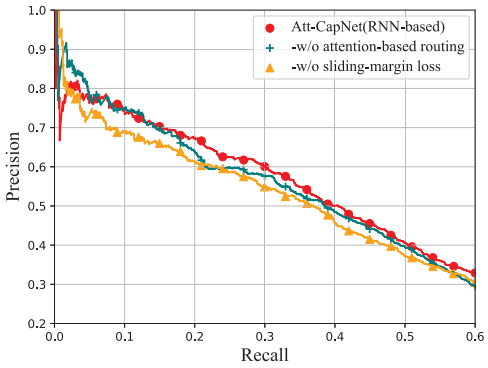

Figure 4: The PR curves of the Att-CapNet (RNN-based) with various modules on NYT-10

Case Study

We present practical cases in NYT-10 test set to show the effectiveness of our feature clustering method (Att-CapNet) compared to max-pooling, linear average (Avg.) and word-level attention (Att.) with the same feature extracting network (RNN). Table 7 shows three multi-labeled sentences for relation extraction by all the four methods from which we can conclude that, 1) Att-CapNet method has recognized all the labeled relations. 2) the other three methods can only give one confident prediction. 3) a few methods even cannot recognize any relations such as max-pooling for the sentence *S1*. 4) our feature clustering method is more capable to recognize highly overlapping relations.

Conclusions and Future Work

In this paper, we propose a novel capsule-based approach for multi-labeled relation extraction to handle the highly overlapping relations and improve the capability of clustering relation features. To our best knowledge, this is the first attempt that applies capsule network to solve the challenging task. The proposed model consists of a concise pipeline. First, we extract low-level semantic information with Bi-LSTM. Then, the low-level features are clustered to be high-level relation representations with attentive capsule network. Finally, sliding-margin loss is proposed to train the model reasonably with all the relations including the "no relation". Our experiments show that the proposed approach achieves significant improvement for multi-labeled relation extraction over previous state-of-the-art baselines.

In future, our solutions of features clustering can be generalized to other tasks that deal with overlapping and discrete features. For instance, a possible attempt might be to perform reading comprehension.

Acknowledgments

This work is supported by National China 973 Project No. 2015CB352401; Chinese National Research Fund (NSFC) Key Project No. 61532013 and No. 61872239. FDCT/0007/2018/A1, DCT-MoST Joint-project No. (025/2015/AMJ), University of Macau Grant Nos: MYRG2018-00237-RTO, CPG2018-00032-FST and SRG2018-00111-FST of SAR Macau, China. National Key Research and Development Program of China (No. 2017YFB0304100), National Natural Science Foundation of China (No. 61672343 and No. 61733011), Key Project of National Society Science Foundation of China (No. 15-ZDA041), The Art and Science Interdisciplinary Funds of Shanghai Jiao Tong University (No. 14JCRZ04).

References

Graves, A. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Ó Séaghdha, D.; Padó, S.; Pennacchiotti, M.; Romano, L.; and Szpakowicz, S. 2009. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations*.

Hinton, G. E.; Krizhevsky, A.; and Wang, S. D. 2011. Transforming auto-encoders. In *Proceedings of the ICANN*.

Hinton, G. E.; Sabour, S.; and Frosst, N. 2018. Matrix capsules with em routing. In *Processings of the ICLR*.

Kim, Y. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the EMNLP*.

Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. In *Processings of the ICLR*.

Lin, Y.; Shen, S.; Liu, Z.; Luan, H.; and Sun, M. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the ACL*.

Liu, C.; Sun, W.; Chao, W.; and Che, W. 2013. Convolution neural network for relation extraction. In *Proceedings of the ADMA*.

Liu, T.; Zhang, X.; Zhou, W.; and Jia, W. 2018. Neural relation extraction via inner-sentence noise reduction and transfer learning. In *Proceedings of the EMNLP*.

Liu, P.; Qiu, X.; and Huang, X. 2016. Recurrent neural network for text classification with multi-task learning. In *Proceedings of the IJCAI*.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the ACL and the IJCNLP*.

Miwa, M., and Bansal, M. 2016. End-to-end relation extraction using LSTMs on sequences and tree structures. In *Proceedings of the ACL*.

Mooney, R. J., and Bunescu, R. C. 2006. Subsequence kernels for relation extraction. In *Proceedings of the NIPS*.

Nguyen, T. H., and Grishman, R. 2015. Combining neural networks and log-linear models to improve relation extraction. *arXiv preprint arXiv:1511.05926*.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the EMNLP*.

Riedel, S.; Yao, L.; and McCallum, A. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the ECML-PKDD*.

Sabour, S.; Frosst, N.; and Hinton, G. E. 2017. Dynamic routing between capsules. In *Proceedings of the NIPS*.

Santos, C. N. d.; Xiang, B.; and Zhou, B. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the ACL*.

Socher, R.; Huval, B.; Manning, C. D.; and Ng, A. Y. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the joint conference on EMNLP and CoNLL*.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.

Surdeanu, M.; Tibshirani, J.; Nallapati, R.; and Manning, C. D. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the Joint Conference on EMNLP and CoNLL*.

Turian, J.; Ratinov, L.; and Bengio, Y. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the ACL*.

Wang, L.; Cao, Z.; de Melo, G.; and Liu, Z. 2016. Relation classification via multi-level attention cnns. In *Proceedings of the ACL*.

Wang, Q.; Qiu, J.; Gao, D.; Gao, J.; et al. 2018. Recurrent capsule network for relations extraction: A practical application to the severity classification of coronary artery disease. *arXiv preprint arXiv:1807.06718*.

Xi, E.; Bing, S.; and Jin, Y. 2017. Capsule network performance on complex data. *arXiv preprint arXiv:1712.03480*.

Xu, K.; Feng, Y.; Huang, S.; and Zhao, D. 2015. Semantic relation classification via convolutional neural networks with simple negative sampling. In *Proceedings of the EMNLP*.

Xu, Y.; Jia, R.; Mou, L.; Li, G.; Chen, Y.; Lu, Y.; and Jin, Z. 2016. Improved relation classification by deep recurrent neural networks with data augmentation. In *Proceedings of the COLING*.

Yang, D.; Wang, S.; and Li, Z. 2018. Ensemble neural relation extraction with adaptive boosting. In *Proceedings of the IJCAI*.

Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J.; et al. 2014. Relation classification via convolutional deep neural network. In *Proceedings of the COLING*.

Zhang, D., and Wang, D. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006*.

Zhang, Z.; Zhao, H.; and Qin, L. 2016. Probabilistic graph-based dependency parsing with convolutional neural network. In *Proceedings of the ACL*.

Zhao, W.; Ye, J.; Yang, M.; Lei, Z.; Zhang, S.; and Zhao, Z. 2018. Investigating capsule networks with dynamic routing for text classification. In *Proceedings of the EMNLP*.

Zhou, P.; Shi, W.; Tian, J.; Qi, Z.; Li, B.; Hao, H.; and Xu, B. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of ACL*.

Zhu, J.; Qiao, J.; Dai, X.; and Cheng, X. 2017. Relation classification via target-concentrated attention cnns. In *Proceedings of the ICONIP*.