SENet for Weakly-Supervised Relation Extraction

ABSTRACT

Sentence relation extraction aims to extract relational facts from sentences, which is an important task in natural language processing(NLP) field. A common pipeline in relation extraction(RE) task is to first encode a sentence into latent representation, and then classify it utilizing a combination of attention mechanism, fully connected (FC) layers and softmax layer. Therefore a good sentence encoder for relation extraction is essential. Previous models use shallow convolution neural networks or recurrent neural networks like bi-directional LSTMs to extract features of a sentence. To enhance the representation power of network, in this paper we investigate the capability of SE module on the task of distantly supervised noisy relation extraction. SE module could learn the importance of each filter, enhance important filters and restrain less important ones, therefore prevent overfitting. To enhance position information, we also demonstrated the power of a new scheme of pooling named double pooling. Experimental results demonstrate the effectiveness of our method compared with baseline models.

KEYWORDS

Natural Language Processing, Relation Extraction, Distant Supervision

1 INTRODUCTION

Relation Extraction devotes to extracting relational facts from sentences, which can be applied to many natural language processing (NLP) applications. Given a sentence with an entity pair e_1 and e_2, this task aims to identify the relation between e_1 and e_1. RE has drawn much attention of many researchers in NLP field. [13] is among the first work to apply neural networks in this task. They adopted Convolutional Neural Network(CNN) to automatically extract the sentence representations with the raw input words for RE, which achieved significant improvements compared with traditional models. [15] applied attention mechanism with Long Short-Term Memory (LSTM) Networks to capture the semantic information in a sentence, which didn't utilize any features derived from lexical resources or NLP systems. However, supervised data relies on human annotation which is very costly. Therefore, obtaining large scale training data remains a major issue.

To obtain large scaled relation extraction dataset, [9]proposed Distant Supervision paradigm to automatically scale RE to large domains. They used the relational facts from large scaled Knowledge Bases (KB) to automatically align with texts. Specifically, for a triplet fact $r(e_1, e_2)$ in a KB, all sentences that mention both entities e_1 and e_2 are aligned with relation r. Figure 1 shows this process. We call the set containing an entity pair with sentences mentioned them as a Bag. We also introduce NA relation, which represents no relation between two entities.

With distant supervised dataset, [12] proposed PCNN to automatically extract features from sentences and applied MIL to select the most important sentence. Both [7] and [4] applied attention mechanism to alleviate the inference of noises. [4] also utilize the entity description as external information to improve performance. [5] used cross-sentence max pooling to take all sentences into consideration and considered the multi-label problem. [6] used cross lingual attention to consider the information consistency and complementarity among cross-lingual texts. [8] used a dynamic transition matrix to characterize the noise and apply curriculum learning framework to guide training.

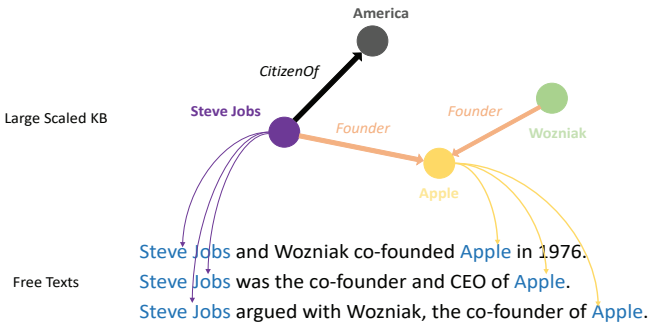

Figure 1: Distant supervision dataset

However almost all of the above model use vanilla CNNs, typically one convolution layer and max pooling as sentence encoder. [2] used nine layers of ResNet and its performance is comparable with PCNN-ATT. In this paper, we investigate the effects of SENet for distantly-supervised relation extraction. In CNN, not all filters could learn important features. SE module could adaptively recalibrate channel-wise feature responses by explicitly modelling interdependencies between channels, enhance those important filters and restrain those are less important, therefore prevent overfitting and improve the representation power of network. We design a convolutional neural network with four SE blocks to extract features of sentences. We evaluate on the NYT-Freebase dataset [11], and demonstrate the state-of-the-art performance. Our contributions are three-fold:

1. We are the first to consider SENet for weakly-supervised relation extraction;
2. We show that out model outperforms all others by a large margin empirically, obtaining state-of-the-art performances;
3. We used double pooling and Swish activation function in our model, achieving a better result.

2 METHODOLOGY

In this section, we describe a novel SENet architecture for distantly supervised relation extraction. To explore the capability of SENet as sentence encoder for RE, we compare our model with other models without utilizing sentence level attention mechanism. Figure 2 shows the architecture of our model. We design this network using four SE-ResNet blocks and double pooling(SE-ResNet-D).

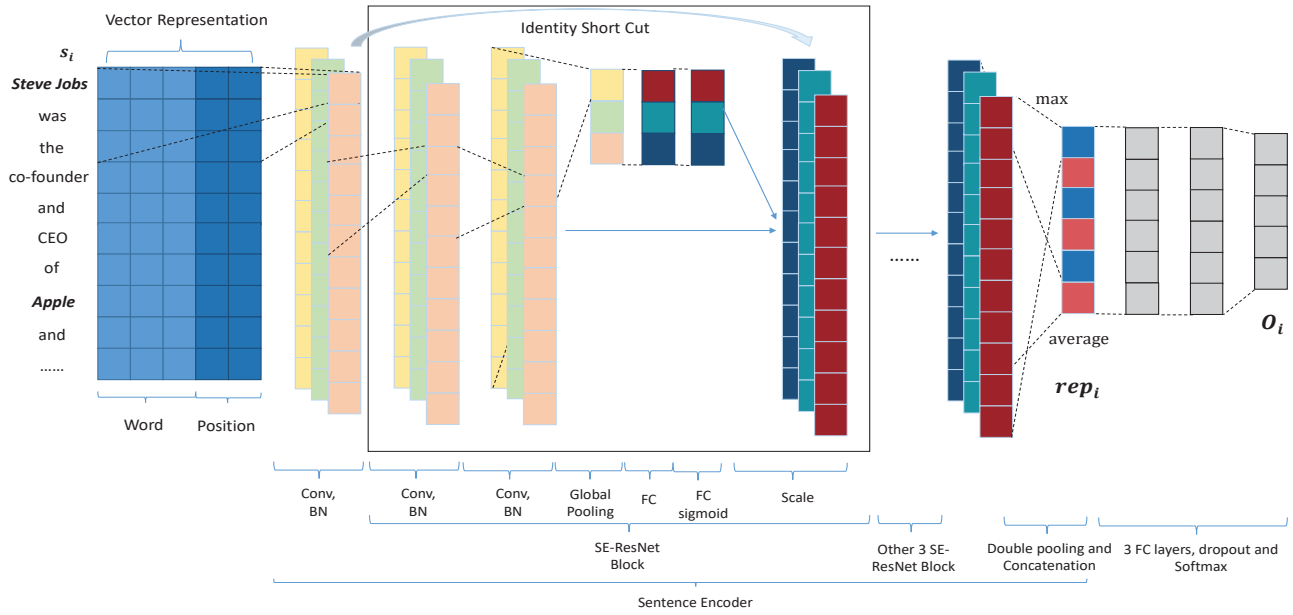

Figure 2: The structure of our model: SE-ResNet-D. Input a sentences$_i$, output the probability distribution O_i

2.1 Vector Representation

Word embeddings are low dimensional vector of tokens, which are learned from the large unlabeled text. We denote the embedding of a word as $\boldsymbol{v_w}$. Position embeddings are low dimensional vectors of positions. The relative position is the distance between token and entity. Every relative position is corresponding to a dense vector $\boldsymbol{v_p}$ which is called position embedding. The vector representation $\boldsymbol{v}$ is concatenated by word embedding $\boldsymbol{v_w}$ and position embedding $\boldsymbol{v_w}$ as shown in the Vector Representation part in Figure 2.

2.2 "Squeeze-and-Excitation"(SE) block

To improve the representational power of a network by explicitly modelling the interdependencies between the channels of its convolutional features, SENet have been proposed. It allows the network to perform feature recalibration, through which it can learn to use global information to selectively emphasize informative features and suppress less.In other word, it is a channel-wise attention mechanism. An SE network can be generated by simply stacking a collection of SE building blocks. Figure 2 shows the architecture of SE-ResNet block. The SE Block is composed of three parts: "Squeeze", "Excitation" and Scaling. We denote $\boldsymbol{U}$ as features from last layer. Formally, after global average pooling, which is the "Squeeze" operation, a statistic $\boldsymbol{z} \in \boldsymbol{R^C}$ is generated by shrinking $\boldsymbol{U}$ through spatial dimensions $H \times W$(in our model $W = 1$), where the c-th element of z is calculated by:

$$z_c = \boldsymbol{F}_{sq}(\boldsymbol{u_c}) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{1}$$

SE Block employs a simple gating mechanism with a sigmoid activation as "Excitation" operation:

$$s = \boldsymbol{F}_e x(z, \boldsymbol{W}) = \sigma(g(z, \boldsymbol{W})) = \sigma(W_2 \delta(W_1 z)) \tag{2}$$

where δ refer to nonlinear activation function (in our paper it is Swish instead of ReLU, which will be illustrated in chapter 3.6), $W_1 \in R^{\frac{C}{r} \times C}$ and $W_2 \in R^{C \times \frac{C}{r}}$ representing 2 fully connected (FC) layers respectively(r is reduction ratio, used to limit model complexity, in our implementation $r = 16$). The final output of the block is obtained by rescaling the transformation output $\boldsymbol{U}$:

$$\widetilde{x_c} = \boldsymbol{F}_{scale}(\boldsymbol{u_c}, s_c) = s_c \cdot \boldsymbol{u_c} \tag{3}$$

where $\widetilde{X} = [\widetilde{x_1}, \widetilde{x_2}, ..., \widetilde{x_c}]$ and $\boldsymbol{F}_{scale}(\boldsymbol{u_c}, s_c)$ refers to channel-wise multiplication between the feature map $\boldsymbol{u_c} \in R^{H \times W}$ and the scalar s_c.

2.3 SE-ResNet Module

SE blocks are sufficiently flexible to be used in residual networks. The black frame in Figure2 depicts the schema of an SE-ResNet module. We apply Batch Normalization (BN)[3] after each convolution layer. Squeeze and excitation both act before summation with the identity branch.

2.4 Pooling and Concatenation

Max pooling over time is good at capturing strong feature wherever it appears while average pooling is good at capturing position information. To take the advantage of both max pooling and average pooling, we proposed a new scheme of pooling named double pooling. We denote the max pooling of vector for each filter as c_{max}^{iq} (i denote ith sentence, q denote qth filter) and the

Table 1: Hyperparameter settings

Hyperparameter	Value
Convolution kernel size k	3
Word dimension d_w	50
Position dimension d_p	5
Position maximum distance e_{max}	30
Position minimum distance e_{min}	-30
Number of filters C	128
Initial Learning Rate λ	0.001
Mini-batch size B	64
Dropout probability p	0.5
Batch Normalization eps ϵ	0.001
L2 regularization λ_{l2}	0.001

average pooling of vector for each filter as c_{ave}^{iq}. We concatenate all these vector to form the representation of sentence $\boldsymbol{rep_i} = [c_{max}^{1q}; c_{ave}^{1q}; c_{max}^{2q}; c_{ave}^{2q}; ...; c_{max}^{nq}; c_{ave}^{nq}]$.

2.5 Swish Activation Function

Swish is defined as $x \cdot \sigma(\beta x)$, where $\sigma(z) = (1+e^{-z})-1$ is the sigmoid function and β is a trainable parameter in our implementation. [10]proposed this novel activation function and proved it effective in many cases. We verifiy this by simply replacing ReLU with Swish in our model and achieved a better result.

2.6 Fully Connected Layers and softmax

After we get the representation $\boldsymbol{rep_i}$ of sentence $\boldsymbol{s_i}$, we apply three FC layers to output the confidence vector $\boldsymbol{O_i}$. Then the conditional probability of j-th relation is

$$p(rel_j|\Theta, \boldsymbol{s_i}) = \frac{e^{o_j}}{\sum_{k=1}^{M} e^{o_k}} \tag{4}$$

3 EXPERIMENTS

3.1 Experimental Settings

In this paper, we use the word embeddings released by [7] which are trained on the NYT-Freebase corpus [11]). The input text is padded to a fixed size of 100. Training is performed with tensorflow adam optimizer. The implementation is done using Tensorflow 1.4.0. In Table 1 we show all hyperparameters used in the experiments.

We experiment with several baselines and variants of our model. CNN + ATT and PCNN+ATT[7] extract sentence features with CNN and both utilized the attention mechanism. PCNN use piese-wise max pooling instead of vanilla max pooling. ResCNN-9 use a shallow ResNet to be the sentence encoder and without attention. BLSTM+ATT[15] use BLSTM as sentence feature extractor and word level attention mechanism. We implement ResCNN-9 and BLSTM by ourselves and build another model BGRU+2ATT, which replace BLSTM with BGRU and use both word level and sentence level attention mechanism.

3.2 NYT-Freebase Dataset

There are 522,611 sentences in training data and 172,448 sentences in testing data in this dataset. Similar to previous work[7][12], we

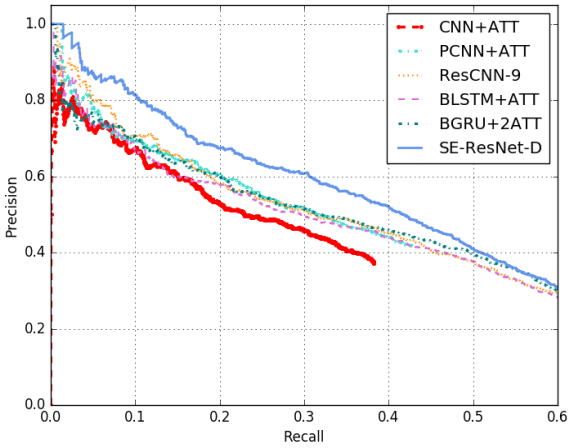

Figure 3: Comparing our model with other baselines

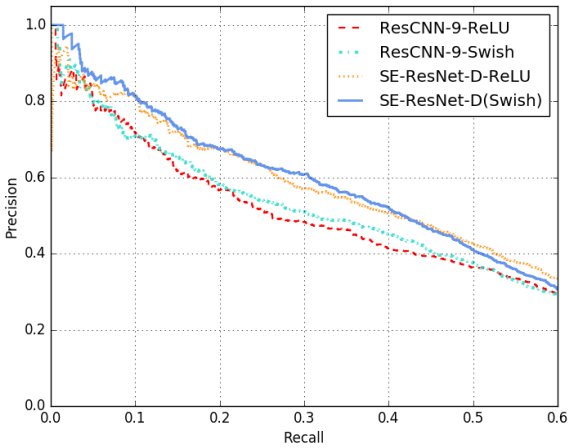

Figure 4: Comparing ReLU and Swish on ResCNN-9 and our model

evaluate our model using the held-out evaluation. Following[14], we simply chose the prediction of the sentence with the highest score (but not NA) as the prediction of the bag. We report the aggregate curves precision/recall curves and Precision@N (P@N).

3.3 Model performance and comparison

In Figure 4, we compare the proposed SE-ResNet-D model with other baselines. We find that our model dominates the precision/recall curve overall. We design the architecture of our model using grid search and find that with four SE-ResNet blocks, we could improve the performance of learning in a noisy input setting. Performance will not get better if we use more SE-ResNet blocks due to the limit size of the NYT dataset. We also show the effect of Swish activation function in Figure 5. Swish is a slightly better than ReLU. Figure 6 shows the effect of SE module and double pooling. In Table2, we

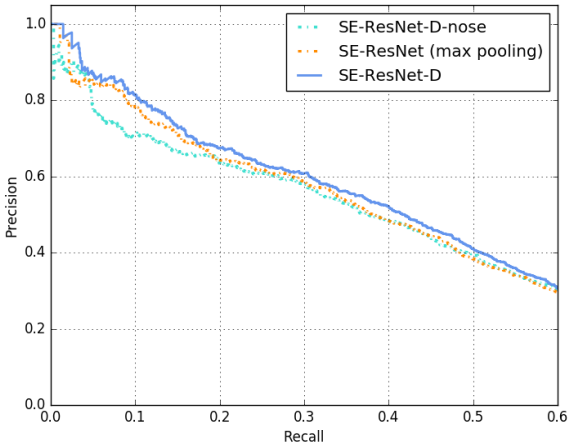

Figure 5: Verify the effect of SE Module and double pooling

Table 2: P@N for relation extraction with different models. Our model outperforms all other state-of-art techniques

P@N(%)	100	200	300	Mean
CNN + ATT	76.2	68.6	59.8	68.2
PCNN + ATT	76.2	73.1	67.4	72.2
ResCNN-9	82.0	73.0	70.7	75.2
BLSTM + ATT	76.0	71.0	66.0	71.0
BGRU + 2ATT	76.0	72.5	67.7	72.1
SE-ResNet-D-ReLU	84.0	81.5	74.3	79.9
SE-ResNet (max pooling)	85.0	80.5	74.3	79.9
SE-ResNet-D-nose	83.0	73.0	69.7	75.2
SE-ResNet-D (our model)	**87.0**	**82**	**77.0**	**82.0**

compare the performance of our models with state-of-art baselines and variants of our model. Even without sentence-level attention mechanism, our model is still ahead of other baselines by a large margin.

The reason of SE-ResNet-D help this task is in three aspects. First, SE module improve the representational capacity of a network by enabling it to perform dynamic channelwise feature recalibration and prevent overfitting. Second, multiple layers of convolution with identity shortcut and SE module extract multi-scale information including hidden lexical, syntactic and semantic representations of a sentence. Finally, comparing with max pooling over time, double pooling could better incorporate position information of a feature map.

4 CONCLUSION

In this paper, we propose a novel architecture for distantly-supervised relation extraction. With SE module, the performances are much improved. These results proved that SENet could better encode information of sentences than those used in other models like vanilla CNNs or RNNs and its variants. We also proved that with double pooling and Swish activation function, the performance could be further improved.

REFERENCES

[1] Jie Hu, Li Shen, and Gang Sun. 2017. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507* (2017).

[2] Yi Yao Huang and William Yang Wang. 2017. Deep Residual Learning for Weakly-Supervised Relation Extraction. *arXiv preprint arXiv:1707.08866* (2017).

[3] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

[4] Guoliang Ji, Kang Liu, Shizhu He, Jun Zhao, et al. 2017. Distant Supervision for Relation Extraction with Sentence-Level Attention and Entity Descriptions.. In *AAAI*. 3060–3066.

[5] Xiaotian Jiang, Quan Wang, Peng Li, and Bin Wang. 2016. Relation extraction with multi-instance multi-label convolutional neural networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. 1471–1480.

[6] Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2017. Neural Relation Extraction with Multi-lingual Attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 34–43.

[7] Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 2124–2133.

[8] Bingfeng Luo, Yansong Feng, Zheng Wang, Zhanxing Zhu, Songfang Huang, Rui Yan, and Dongyan Zhao. 2017. Learning with noise: enhance distantly supervised relation extraction with dynamic transition matrix. *arXiv preprint arXiv:1705.03995* (2017).

[9] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 1003–1011.

[10] Prajit Ramachandran, Barret Zoph, and Quoc V Le. 2018. Searching for activation functions. (2018).

[11] Limin Yao, Sebastian Riedel, and Andrew McCallum. 2010. Collective cross-document relation extraction without labelled data. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1013–1023.

[12] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 1753–1762.

[13] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. 2335–2344.

[14] Xiangrong Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Large Scaled Relation Extraction with Reinforcement Learning. *Relation* 2 (2018), 3.

[15] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vol. 2. 207–212.