Research on Relation Extraction Method Based on Similar Relations and Bayesian Neural Network

Yiqiu Fang¹, Yingjie Zang¹* and Junwei Ge²

¹Computer science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, Chongqing, 400000, China
²Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing, Chongqing, 400000, China

*Corresponding author’s e-mail: zhangyingjie667@gmail.com

Abstract. The solidification of network parameter values for most relation extraction models after training makes the model overconfident in the process of prediction and classification tasks. Moreover, the interference of similar relation in the text data will affect the effect of relation extraction. We propose a relation extraction model based on relation similarity and Bayesian neural network. This model uses the logistic regression loss function to make the training parameters closer to the target relation and away from the similar relation, thereby eliminating the interference of the similar relation in the relation extraction task. In addition, it is also possible to learn a probability distribution from the weights of the Bayesian-LSTM (Bayesian-Long Short-Term Memory) neural network, and retain the corresponding uncertainty on the basis of obtaining long-distance dependence information using LSTM, so that the model learns more data features while performing regularization at the weight level. The model also uses an attention mechanism to pay more attention to useful information. The experimental results on the Wikipedia data set TACRED (TAC Relation Extraction Dataset) data set show that the proposed method effectively improves the effect of the entity relation extraction model.

1. Introduction

Relation extraction is mainly to extract the corresponding relation between entity pairs from scattered, complex, and unstructured text. Its manifestation is the <head, relation, tail> triplet. Relation extraction is a subtask of information extraction, and it is based on named entity recognition. It is of great significance for tasks such as constructing knowledge graphs, question answering systems and information retrieval.

The remote supervision method proposed by Mintz[1] has milestone significance in relation extraction tasks. This method replaces manual annotation of corpus by aligning the knowledge base. However, remote supervision will bring a lot of noise. Many scholars have conducted research on noise reduction work. A typical example is Surdeanu[2] who proposed a multi-label and multi-instance extraction method for probabilistic graph models. With the rapid development of deep learning, deep learning models have achieved good results in various fields. In entity relation extraction tasks, deep learning models are gradually being applied. Zeng[3] first proposed the use of CNN for relation extraction in 2014, using Convolutional Deep Neural Network (CDNN) to extract vocabulary and sentence-level features. Zhou[4] proposed the Bi-directional Long Short-Term Memory sequence model and the attention mechanism to extract entity relation, and achieved good results.
extraction tasks are also applied to many other fields, such as the use of two-way simple recurrent neural network and convolutional neural network with attention mechanism to extract the relation between chemical substances and diseases[5].

Although deep learning models perform well in the task of relation extraction and improve the accuracy of entity relation extraction, these models will suffer from model failure due to many parameters and high model complexity and small amount of training data during training. We combines the relation similarity measurement method with Bayesian-LSTM model. The similarity distance is maximized by the loss function to avoid the model predicting the entity relation as a similar relation instead of the actual entity relation. At the same time, uncertainty is introduced. Bayesian-LSTM learns the probability distribution in the weights of the neural network through the Bayesian Backpropagation method, so that the network has a regularization effect. In this way, the generalization ability of the entity relation extraction method can be improved, and better results can be achieved in the entity relation prediction task.

2. Based on relation similarity and Bayesian neural network model

The relation extraction method based on relation similarity and Bayesian neural network proposed includes obtaining the similar relation of entities in the data set through Bayesian multilayer perceptron, and learning the uncertainty of weights through Bayesian long-term short-term memory network. The attention mechanism be used to integrate network learning content.

First the original input be converted into a word vector representation, and then put the obtained word vector as the input of Bayesian-LSTM to learn the dependency information of the sentence, and through a Gaussian distribution sampling with a mean $\mu$ and a variance $\rho$ obtain the initial parameters as the prior probability. At the same time, put the word vector into Positional Embedding to get the position information. Taking the obtained position information and sentence dependency information as the input of the attention mechanism, and integrate it into “Final_atten”. Secondly, the label data set of triples containing head entity, tail entity and relation trained on the Bayesian multilayer perceptron model, and the similarity between relation be calculated through KL divergence[6]. Finally, the calculated relation similarity is added to the relation extraction model.

![Figure 1: Relation extraction model based on similarity relation and Bayesian neural network](image1)

![Figure 2: Bayesian-LSTM model](image2)

2.1. Word vector and location information representation

We adopt the dense word vector representation method of Mikolov[7]. Each word is mapped to a word vector of a certain dimension, and similar words are represented as similar vectors. The vectors are no longer unrelated, but contain rich semantic information. Because this method has achieved good results in deep learning tasks, most word vector representations currently use dense word vector
methods. The sentence converted in the text to \( S_i (w_i, w_j, \text{L} , w_k) , j \in \text{input} , S_j \) means the \( j \) th sentence in the input sentence, \( S_j \in \mathbb{R}^{d_{w}} \), \( d_{w} \) means the number of words in the sentence, \( d_{w} \) means the dimension of the word vector, \( w_i \) means the \( n \) th word vector in the sentence, the dimension is \( d_{w} \).

In the task of relation extraction, the closer the entity is to the relation, the greater the contribution to the relation, so we adopt the location information extraction method proposed in \[3\]. The relative distance be calculate between the current word and other words to get the location information. For example the sentence “The bombing did not affect my uncle's twins, Hans and Tom”, the distance between the relation twins and the entity my uncle's is \( d_{head} = 1 \), and the distance to the entity Hans \( d_{tail} = 1 \). The position information from the current word to the two entity words is obtained: \( x_{\text{ind}} = [d_i, d_j] \), \( i \) indicates the current word.

2.2. Bayesian model and long short-term memory network

Bayesian neural networks can regularize the model at the weight level, reducing the model’s overconfidence and overfitting in prediction and classification tasks. Secondly, more representations can be learned through models with fewer parameters and lower complexity\[8\]. We introduce Bayesian-LSTM\[9\], a Bayesian long and short-term memory network, which not only learns the dependent information in sentences through LSTM, but also obtains richer representations through the uncertainty distribution of learning weights through the characteristics of Bayesian neural networks. Compared with ordinary neural networks, the weights of Bayesian neural networks sample the weights through a probability distribution, and then the probability distribution parameters can be optimized. The following is an example to illustrate the role of Bayesian network in relation extraction task.

Example: (1) The bombing did not affect my uncle’s twins, Hans and Tom.
(2) Hans and Tom are my uncle’s twins, survived the explosion.

When traditional LSTM models perform prediction tasks after text training, there will be many different changes due to the solidification of parameters and the structure of the composed sentence, resulting in insufficient predictive ability for sentences whose structure is easily changed. For example, in sentence (1), Hans is not affected by the explosion and Hans in sentence (2) means that Hans survived the explosion, but if you use traditional fixed parameter LSTM training sentence (1), and use sentence (2) as test sentences, prediction and classification accuracy will be very low. Therefore, we use a probability distribution that obeys the normal distribution to retain the uncertainty of the parameters, so that it has better performance in the face of changing sentence structure.

2.2.1. Bayesian-LSTM

The Bayesian long and short-term memory network neural network used in this article introduces a priori hypothesis, so the Bayesian network probability model is \( E_{\psi \theta} P(\gamma | x, \theta) \). \( \theta \) is the parameter value in the model LSTM, \( r \) is the relation that needs to be extracted from the text, and \( x \) is the text input into the model. Using the idea of variational posterior\[10\], formula (1) is train the parameters \( \theta \) of the model. The symbol \( \sigma \) represents the sigmoid function.

\[
F(\theta) = \sum_{r \in R} KL[ q(\theta) \| P(\theta)] - E_{\psi \theta}[\log P(\gamma | x, \theta)]
\] (1)

The vector representation of the sentence obtained from the embedding layer is used as the Bayesian-LSTM input, and each Bayesian-LSTM layer outputs a state \( h_i \). The state \( h_i \) and the input vector representation \( x_{\text{ind}} \) are used as the input of the next Bayesian-LSTM, and then the state \( h_{i+1} \) is generated. Analogy until the last time step Bayesian-LSTM unit. As a sequence model, Bayesian-LSTM not only captures the context-dependent information in the text, but also uses the characteristics that the weight is not fixed and obeys the probability distribution to retain the uncertainty for the model. The final output of this layer is equation (2)

\[
(h_1, h_2, \text{L}, h_n) = \text{Bayesian-LSTM}(S_1, S_2, \text{L}, S_n)
\] (2)
2.2.2. Attention mechanism

The attention mechanism has achieved very good results in the application of various fields of deep learning, including image processing, speech processing, natural language processing and other fields. Bahdanau\(^{[11]}\) and others applied the attention mechanism to natural language processing for the first time. In the task, good results were achieved in the translation model. This method is similar to that when human beings pay attention to text, they pay more attention to the places of interest, and obtain important information from a large amount of text information. The attention position information \(p_{pos_i}\) and Bayesian-LSTM state information \(h_i\) are used as input, calculated by the following attention formula:

\[
a_i = \text{softmax}(w_a \cdot \tanh(w_p \cdot p_{pos_i} + w_h \cdot h_i))
\]

\[
\text{Final\_atten} = \sum_{i=1}^{n} a_i h_i
\]

2.3. Relation similarity

We draw on Chen\(^{[6]}\)’ relation similarity measurement method and improve it. It uses Bayesian Multilayer Perceptron Bayesian-MLP to obtain the vector of the entity and its corresponding relation. Finally, the KL divergence method is used to measure the similarity between entity relation. The similarity is introduced in the relation extraction task, and the loss function is used to maximize the distance between the entity relation in the training set and the corresponding similar relation obtained, which avoids the interference problem of the similar relation in the relation prediction task, and can improve the model’s performance.

Example: The bombing did not affect my uncle's twins, Hans and Tom.

When the traditional relation extraction model performs relation prediction, the relation between Hans and Tom is easily judged as parents instead of brother. This is the prediction error caused by the similar relation, which leads to the low accuracy of relation prediction. Therefore, this article introduces the relation similarity to strengthen the judgment of similarity.

2.4. Output layer

The logistic regression loss function be use to maximize the distance between the predicted relation and the similar relation, and minimize the distance between the predicted relation and the target relation, so that the similar relation will not interfere with the correctness of the relation prediction as much as possible when predicting the relation. Relation prediction errors caused by similar relation.

\[
J(\theta) = J(\theta_1) + J(\theta_2) = \sum_i \log p_o(r^{(0)}; \theta_1) - 2 \log p_{\text{label}}(r^{(0)}; \theta_2) + \sum_i \log p(r^{(0)}; \theta_2) - 2 \log p_{\text{label}}(r^{(0)}; \theta_2)
\]

\[
p_o(r^{(0)}; \theta_1) = p(r^{(0)} = r^{(0)} | r^{(0)}; \theta_1) = 1 - p(r^{(0)} \neq r^{(0)} | r^{(0)}; \theta_1)
\]

\[
p_{\text{label}}(r^{(0)}; \theta_2) = p(r^{\text{label}}(0) = r^{(0)} | r^{(0)}; \theta_2) = 1 - p(r^{\text{label}}(0) \neq r^{(0)} | r^{(0)}; \theta_2)
\]

\[
J(\theta_1) \text{ indicates the loss function of the prediction relation } r^{(0)} \text{ and the label relation } r^{\text{label}}(0), \text{ } J(\theta_2)
\]

indicates the loss function of the prediction relation \(r^{(0)}\) and the similar relation \(r^{(s)}\), set the distance between \(r^{(0)}\) and \(r^{\text{label}}(0)\) to 1, set \(r^{(0)}\) and \(r^{(s)}\)’s distance is set to 0, by logistic regression function. \(\theta\)

Indicates that the learnable parameters of \(\theta_1\) and \(\theta_2\) are included.

3. Experiment

We select the Wikidata data set and the TACRED data set, using Wikidata as the data for training the relation similarity model, and TACRED as the data for training the relation extraction model. Wikidata is provided by literature\(^{[12]}\), a one-stop knowledge graph data management platform launched by the Wikipedia Foundation in 2012. Currently Wikidata contains 25 million entity data, and the total number of triples has reached 140 million. We select 1160 relation words, 112946 entity words, 423991 triples from the Wikipedia data set as the training set, and 6065 triples as the test set. We select the TACRED data set in the LDC corpus\(^{[13]}\) in the relation extraction model part. These data
come from the news line and network text in the corpus used in the annual TAC KBP competition. There are 68,124 examples in the training set and 22,631 examples in the test set.

3.1. Bayesian-LSTM and Similarity impact analysis

We propose to add Bayesian-LSTM, which can learn the long-distance sequence dependence information and the weight probability distribution, into the relation extraction model, and the initialization of the weight be set as the prior probability, and then the network weight obtained by the variational posterior method. In order to verify the effectiveness of the Bayesian-LSTM method, we have designed three groups of methods based on PA-LSTM relation extraction for comparison. One group is the comparison between the original PA-LSTM model and the PA-Bayesian-LSTM model, and the second group is the original The PA-LSTM model is compared with the PA-LSTM-SIM model. The third group is the PA-Bayesian-LSTM model and the PA-Bayesian-LSTM-SIM model. Table 2 shows the evaluation values of the first two models.

From Table 2, it can be seen that the accuracy of the PA-Bayesian-LSTM model is increased by 2.49% compared to the original PA-LSTM model without Bayesian method. The recall rate increased by 0.95%, and the comprehensive evaluation value increased by 1.6%. Figure 3 shows the performance of the comparison model on the data set. The PA-Bayesian-LSTM model learns the weight probability by adding the prior weight and the variational posterior method. Through this weight probability, we can learn more abundant features in a simpler model and make the learned features diversified. The precision rate and recall rate have been improved.

| Model                  | Precision | Recall  | $F_1$ score |
|------------------------|-----------|---------|-------------|
| PA-LSTM                | 63.594%   | 56.751% | 59.978%     |
| PA-Bayesian-LSTM       | 68.564%   | 63.693% | 66.038%     |

Table 2 shows the comparison results of the second set of experiments. On the basis of the original PA-LSTM experiment, the similarity mechanism is added, and the target entity relation and the similar relation are separated through the loss function to improve the accuracy of relation prediction.

| Model                  | Precision | Recall  | $F_1$ score |
|------------------------|-----------|---------|-------------|
| PA-LSTM                | 63.594%   | 56.751% | 59.978%     |
| PA-LSTM-SIM(Logistic-Loss) | 66.084%   | 57.708% | 61.612%     |

Table 3 compares with other models.

| Model                  | Precision | Recall  | $F_1$ score |
|------------------------|-----------|---------|-------------|
| PA-LSTM                | 63.594%   | 56.751% | 59.978%     |
| PA-LSTM-SIM (Softmax-Margin Loss) | 68.164%   | 63.693% | 65.852%     |
| PA-Bayesian-LSTM-SIM (Logistic-Loss) | 68.225%   | 64.937% | 66.540%     |
3.2. Comparative analysis of model results

The entity relation extraction method is compared with PA-LSTM and PA-LSTM-SIM (softmax-margin loss) two classic algorithms on the same data set. The results are shown in Table 4 and Figure 5. It can be seen from Figure 5 that compared with the PA-LSTM model, after the P-R curve stabilizes, the accuracy and recall rate of this model are both greater than that of PA-LSTM. Our model is relatively similar to the PA-LSTM-SIM (softmax-margin loss) model curve. When the recall rate is less than 0.2, the accuracy of the PA-Bayesian-LSTM-SIM (Logistic-Loss) model is much greater than that of PA-LSTM-SIM (softmax-margin loss) accuracy rate. When the recall rate is greater than 0.2, the PR curve of PA-Bayesian-LSTM-SIM (Logistic-Loss) is still above PA-LSTM-SIM (softmax-margin loss), and the comprehensive evaluation value of the model Compared with the PA-LSTM model, it has increased by 5.87%, and compared with the PA-LSTM-SIM (Softmax-Margin Loss) model, it has increased by 0.68%.

4. Conclusion

This paper analyzes the shortcomings of existing relationship extraction methods, and proposes entity relationship extraction methods based on similar relationships and Bayesian neural networks. The logistic regression loss function is used to process the target relationship and the similarity relationship, eliminate the interference of the similarity relationship in the relationship extraction, and use the probability distribution learned by the Bayesian neural network on the weight, so that the network can learn more features. Thereby it performs better on relation extraction tasks. The experimental results
show that the model proposed in this paper has a significant improvement over the traditional relation extraction model. The next step will continue to explore ways to eliminate the influence of similar relationships, such as constructing a relationship forest based on the context of similar relationships, and so on to improve the effect of the model.

Acknowledgment
This work has been generously supported by Application and demonstration of cross-media big data in all media industry and by Grant No. cstc2017zydx-zdx0046 Key industry common key technology innovation special project (major theme special project). We thank the help from Liu Fei and Li Shuai during the completion of the paper.

References
[1] Mintz M., Bills S., Snow R. et al. (2009) Distant supervision for relation extraction without labeled data. Proc 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. Singapore. ACL. 1003-1011.
[2] Surdeanu M., Tibshirani J., Nallapati R. et al. (2012) Multi-instance multi-label learning for relation extraction. Proc of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Stroudsburg. ACL. 455-465.
[3] Zeng D, Liu K, Lai S. et al. (2014) Relation classification via convolutional deep neural network. Proc of the 25th International Conference on Computational Linguistics. Lisbon. ACM. 2335-2344.
[4] Zhou Peng, Shi Wei, Tian Jun, et al. (2016) Attention-based bidirectional long short-term memory networks for relation classification. Proc of the 54th Annual Meeting of the Association for Computational Linguistics. ACL. 207-212.
[5] Cao Chunping, He Yaji. (2020) A method for extracting the relation between chemical substances and disease by integrating BSRU and ATT-CNN. J. Journal of Chinese Computer Systems. 41(04). 794-799.
[6] Chen W, Zhu H, Han X. et al. (2019) Quantifying Similarity between Relations with Fact Distribution. Artificial Intelligence. ACL. 2882-2894.
[7] Mikolov T, Chen K, Corrado G, et al. (2013) Efficient estimation of word representations in vector space. J. arXiv preprint arXiv.1301.3781.
[8] Charles B, Julien C, Koray K, et al. (2015) Weight Uncertainty in Neural Networks. In Proceedings of the 32nd International Conference on Machine Learning., ICML. 1505.
[9] Fortunato M, Blundell C, Vinyals O. (2017) Bayesian Recurrent Neural Networks. J. Preprint arXiv. 1704.02798.
[10] Graves A. (2011). Practical variational inference for neural networks. In Advances in Neural Information Processing Systems (NIPS).2348–2356.
[11] Bahdanau D, Cho K, Bengio Y. (2014). Neural machine translation by jointly learning to align and translate. J. arXiv preprint arXiv.1409.0473.
[12] Denny V, Markus K. (2014) Wikidata: A Free Collaborative Knowledge Base. J. Communications of the ACM. 2335-2344.
[13] Zhang Y, Zhong V, Chen D, et al. (2017) Position-aware attention and supervised data improve slot filling. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. 35-45.