

The Advance of Deep Learning Based Named Entity Recognition

Wenxuan Li*

Queen's University, Kingston, Canada

*Corresponding author: 2016122615@jou.edu.cn

Abstract. Named Entity Recognition is a well-known research direction in the area of deep learning. It takes an essential role in natural language processing. The goal of the Named Entity Recognition is to identify and separate the named entities, such as a person, location, or name, from the entire text. In addition, the deep learning model has achieved remarkable achievements in many other areas, and the deep learning-based named entity recognition method has reached an F score of over 90. The paper summarizes the development of named entity recognition and puts forward a recurrent neural network-based named entity recognition algorithm. The result shows that improving the performance of the Named Entity Recognition model by simply enriching the number and variety of the input datasets or providing the model with substantial computing resources for training is nearly impossible without a significant breakthrough. There still requires another way to improve the NER model in future research.

Keywords: Named Entity Recognition, Natural Language Processing, Deep Learning, Machine Learning

1. Introduction

Named Entity, according to the explanation from automatic content extraction, has three forms of entity mention. It usually refers to PER, LOC, ORG, TIME, and NUM expressions (including time, date, currency, and percentage). Named Entity Recognition (NER) is one of the fundamental research directions in Natural Language Processing (NLP). The research aims to identify the boundary and categorization of the name entities, including PER, LOC, ORG, and so on, inside the regular text to provide support for the downstream tasks, as shown in Figure 1. During the current stage, the research on NER has reached a competitive level. The F score in data set CoNLL 2003 has reached 93.5, and the F score in data set Ontonotes v5 has reached 89.71 [1,2]. Many NER tools have been published online for free download and use.

[**PER** Pierre Vinken] , 61 years old , will join
[**ORG** IMB] 's board as a nonexecutive director
[**DATE** Nov.29] .

Figure 1. The example of NER

It has been acknowledged that the research of NER has gone through a series of models, from the ones based on the rules and regulations (such as the Hidden Markov Model, Maximum Entropy, and Conditional Random Field) to the ones found in the deep learning, just like all the other tasks in NLP. Besides, since the massive advantage of deep learning is illustrated in other areas, the accuracy of deep-learning-based NER has been significantly higher than the accuracy of the traditional statistics-based models in most public data sets (such as CoNLL 2003, Ontonotes v5). However, accuracy cannot be the only measurement of the models. The deep learning-based NER model is more potent in fit theory than the traditional machine learning model due to its high number of variables. The high number of variables could be the reason for the accuracy of NER.

More proofs shows that the deep learning model have achieved high accuracy by learning the specific model-based data sets instead of learning to understand the task itself. Under this condition, the deep learning model can quickly reach high accuracy in a relatively simple task, like classification.

It has a more powerful fitting ability and can acquire enough data from the data sets. However, for those complex missions requiring a higher accuracy, which requires a better understanding of the mission itself, it becomes nearly impossible to use the deep learning model as the final solution during this stage.

The deep learning model seems to hit a plateau as the research progresses. The improvement of the model's efficiency relies solely on the massive number of data sets and the higher computation resources, such as the pre-training method like BERT and GPT-3. However, the data set is still a considerable cost for most of the tasks in NLP, and the model's performance is still below the expectation of the actual application. In the meantime, the deep learning model still has some problems with its terrible interpretability and robustness (adversarial examples technology [3]).

This paper introduces the tasks of NER and the current NER model, which is based on deep learning in section two. Section three illustrates the regular baseline model, including RNN, GRU, LSTM+CRF based on the CoNLL2003 dataset. Section four provides the analysis and discussion of the experiment result. Section five is the conclusion.

2. Related work

Deep learning-based NER is usually considered a sequence tagging task. Such as the sentence shown in figure 2, “Pierre Vinken, 61 years old, will join IBM’s board as a non-executive director Nov.29”. While performing the NER on this sentence, the expectation is to tag the ‘PER’ label onto the named entity “Pierre Vinken” instead of recognizing the “Pierre” and “Vinken” as two different entities. That explains why the BIOES labeling method, including BIOES, is required to separate the entire entity phrase. BIOES Labeling Mode (B-begin, I-inside, O-outside, E-end, S-single): B represents that the word is at the beginning of the expression(begin), I presents the inside, O represents the outside, E represents that the word is at the end of the expression, S represents the word itself can be a phrase. BIO labeling can be considered as a simple version of BIOES labeling.

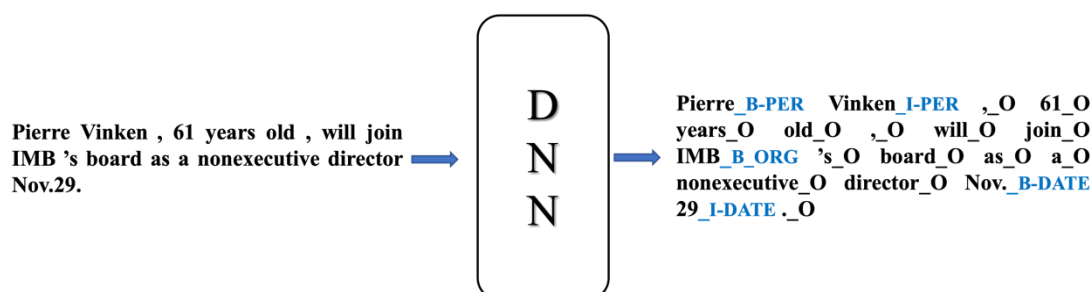


Figure 2. Sequence tagging of the name entity in Chinese and English texts

To be specific, in a phrase such as “Pierre Vinken,” Tag the headword,” Pierre,” inside the phrase with label B: tag the following word in the name entity phrase, “Vinken,” with label I: tag all the remaining term in the sentence with label O. When the entire NER process is finished, the entire sentence should be separated and recorded as “B-X,” “I-X,” and “O.” “B-X” represents the word is the headword of the named entity and belongs to type “X.” “I-X” represents phrase which the word belongs to is the remaining part of the named entity and is in the type of “X.” “O” represents the phrase does not belong to any of the named entities.

There have been plenty of deep learning-based NER models published since the beginning of the 21st century. Some typical representations are the NER models based on the combination of Long Short-Term (LSTM) and Conditional Random Field (CRF) from Lample, F score has reached 90.04 in the CoNLL 2003 dataset [4], and the NER model from Ma, which uses the letter-level word coding characteristic of Convolutional Neural Networks and combines it with the word-level illustration to finish the NER based on LSTM and CRF. As the letter-level characteristic of the word has been considered and analyzed, the F score of the model has reached 91.21 in CoNLL 2033 dataset [5].

To solve the insufficient number of training data sets, Yand came up with an idea of a NER model based on transfer learning. The model has combined the NER training results in English, Spanish,

and Dutch and has achieved the F score of 91.26 in CoNLL 2003 dataset [6]. There is also a neural network tagging NER model proposed by Ye based on phrase recognition instead of word recognition, which performs the NER relaying on the phrase of the text (multi-words entity) and the information of the text, which acquires an F score of 91.38 in the dataset [7]. According to the above research achievements, there is an argument saying that the deep learning-based model shows a huge advantage in learning a variety of characteristics, which makes the human-designed characteristic less valuable. Wu has proposed a solution that proves the reasonable use of the human-designed characteristics could improve the efficiency of the deep learning-based NER model. By importing different word characteristics, like the part of speech (POS) tagging and the form, Wu's model acquired an F score of 91.87 in CoNLL 2003 [8]. In 2018, BERT [9], a pre-training model, became the most popular NPL model. Based on the use of the transformer training [10] and the utilization of a large number of training datasets, it has an excellent performance in multiple tasks in this area. The F score of the BERT-Base model and the BERT-Large model have reached 92.4 and 92.8, respectively, in the CoNLL 2003 dataset. After that, Akbik has developed a solution that combines the hidden state in BiLSTM with the traditional code word embedding. As the use of BiLSTM will link the word embedding with the meaning of the context, it will significantly solve the issue of polysemy of the word.

Meanwhile, the model can dynamically code the word embedding to fit well into the following tasks. The model has reached an F score of 93.09 in the CoNLL 2003 dataset. There is also a model proposed by Baevski, which, based on the BERT, has born an F score value of 93.5 in the CoNLL 2003 dataset [12, 13].

According to the above research, the deep learning-based NER has performed a good accuracy, and the average value of F score is above 90. However, the improvement in the efficiency of the NER model is a long-term slow process (The F score value has improved from 90.94 to 93.50 between 2016 and 2019). Even though the BERT model uses a considerable number of datasets, which also costs vast computing resources, it can only bring limited improvement to the efficiency of the NER model. Thus, the research in NER still has a challenging future.

3. Methods

This paper proposed the NER model, which uses the recurrent neural network (RNN), including basic RNN, LSTM, Gated Recurrent Unit (GRU), and CRF. It has also been trained and tested on CoNLL 2003 dataset. The following is the introduction of the model described and used in the article. RNN refers to a neural network that uses a data sequence as input and performs recursion in the direction of series evolving. All the nodes should be connected in the form of a linked list. The basic definition is shown in Equation 1.

$$S_t = f(U * X_t + W * S_{t-1}) \quad (1)$$

Inside the equation, X_t represents the current input of t RNN. S_{t-1} refers to the output of the RNN at the last moment. S_t represents the RNN output at the current moment. Figure 3 illustrates a simple RNN. As shown in figure 3, when model w is removed from the system, RNN becomes downgraded to a regular neural network. So compared to a random neural network, RNN can take the context of sequence data into consideration, which shows a significant advantage in dealing with sequence data, such as text. In the current stage, popular deep learning frames, such as TensorFlow (Keras.layers.GRU ()) and PyTorch (nn. RNN ()), have provided the user with an easy-accessed RNN access port.

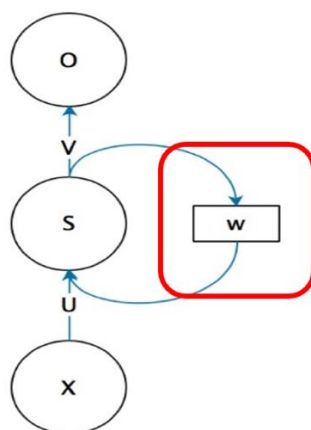


Figure 3. A simple RNN illustration

In some complex situations, predicting the attribute of the word requires consideration of the other words far from the current one. For sequence text, “I grew up in China...I speak fluent Chinese”, The prediction of the word “Chinese” requires consideration of the word “China.” Even though the RNN can handle a dependent situation like this one, it still does not perform well in real-life applications. LSTM is an improvement on RNN. Adding input, output, and forget gates, which require three trained weighted matrixes, to control the input and output of the neural network, then solve the long-term dependency issue. In general, LSTM can be divided into three different stages: 1) forget stage (which performs selective forget on the input from the last node), 2) remember stage (selectively remember the current input), 3) output stage (selectively output the current results). As useless information has been forgotten selectively, LSTM can transfer a more extended sequence of sufficient data.

Though LSTM has a better performance in handling the long-term dependency issue, the difficulty of the model training has also been upgraded due to the increase in the extra variables required. GRU has been regularly used as it requires fewer data and only has reset and update gates to control data in and out from the model. The access ports of GRU in major deep learning frames: Tensorflow (`keras.layers.recurrent.GRU()`) and pytorch(`nn.GRU()`)

CRF is a structured probabilistic model used to label, combine, and classify the structural sequence data. CRF performs a conditional probability $P(Y|X)$ instead of a joint probability distribution $P(X, Y)$ to describe the model itself for the provided output label sequence Y and the watch sequence X . RNN (including basic RNN, LSTM, GRU) model, combined with the CRF, can be used to learn the characteristics of the context of the label, which have a significant effect for NER as RNN can only attract the context of the features. The access ports of the CRF in major deep learning frames: tensorflow (`tensorflow.addons.text.crf`) and Pytorch.

4. Experimental results and analysis

CoNLL 2003 Dataset includes 1393 news articles and 909 Germany new articles. In CoNLL 2003 Dataset, named entity has been separated into four basic types: 1) LOC (location), 2) ORG (organization), 3) PER (person), 4) MISIC (miscellaneous). The model's data set division and parameter setting are shown in Table 1 and Table 2. This paper separates the data set into three functional parts: training, development, and testing. The use of training part is to prepare the model with basic acknowledgment and how this NER model works. The development part is used to adjust the parameters. The testing part is used to conduct a complete evaluation on the model. Besides, the paper initializes the model and the Adam algorithm by glove to perform gradient descent. The batch size of the training set is 128, Epoch is 20, and the probability of Dropout is set to 0.5.

Table 1. Experimental dataset partition

Dataset	Train	Dev	Test
CoNLL 2003	204,567 (23,499)	51,578 (5,942)	46,666 (5,648)

Table 2. Experimental Variables

Word embedding	glove.6B.100d
Optimizer	Adam
Batch size	128
Epoch	20
Dropout	0.5

This paper tests the performance of three deep learning models in NER, namely BiRNN+CRF, BiLSTM +LSTM, and BiGRU+CRF. According to Figure 3, all three outcomes have reached the expectation. The average volume of F score is higher than 80. It also demonstrates that the effectiveness of the LSTM and GRU models outperforms that of the RNN model. Meanwhile, the concurrent rate of the GRU is higher than the anticipation. Still, the LSTM model has a lower accuracy than the data acquired from the article [4]. The possible reasons are the limit on the device's performance, the laptop used here, and the Epoch value has been set to 20, causing the model to end in an incomplete convergent status, which should be the primary reason.

In summary, there has been a basic acknowledgment of the NER and could be able to implement the NER model on the deep learning framework (both TensorFlow and PyTorch have provided easily accessible ports). Meanwhile, there are many implementations tutorial of the BiLSTM + CRF model, which only require the modification of a few lines of code based on the pre-set inside TensorFlow or PyTorch, such as BiRNN+CRF, BiGRU+CRF, BiLSTM +CNNs+CRF.

Table 3. Experiment Result

Model	F1
BiRNN + CRF	82.38
BiLSTM + CRF	82.64
BiGRU + CRF	83.10

5. Conclusion

This paper introduces the deep learning-based named entity recognition model. It analyzes its advantage, and the problems urged to solve. In general, the pros and cons of the NER are the same as the pros and cons of deep learning. In recent years, researchers have put their effort into the combination of rules, characters, and traditional machine learning with deep learning to help modify the learnability and explainability of the deep learning model. According to the statistics result, the portion of the research on reducing the dependency on the labeled datasets, the use of the retrieval models, and the interpretable NLP have been increased. It seems that the trend also affects the area of Nature Language Processing. As a researcher, deep learning is only the solution to artificial intelligence at the current stage. Further research could combine the NER with other traditional methods or make a breakthrough using the frame of deep learning.

References

- [1] Baevski, A., Edunov, S., Liu, Y., Zettlemoyer, L., & Auli, M. (2019, November). Cloze-driven Pretraining of Self-attention Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 5360-5369).

- [2] Akbik, A., Blythe, D., & Vollgraf, R. (2018, August). Contextual string embeddings for sequence labeling. In Proceedings of the 27th international conference on computational linguistics (pp. 1638-1649).
- [3] Zhang W E, Sheng Q Z, Alhazmi A, et al. Adversarial attacks on deep-learning models in natural language processing: A survey[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2020, 11(3): 1-41.
- [4] Lample G, Ballesteros M, Subramanian S, et al. Neural Architectures for Named Entity Recognition[C]//Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016: 260-270.
- [5] Ma X, Hovy E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 1064-1074.
- [6] Yang Z, Salakhutdinov R, Cohen W W. Transfer learning for sequence tagging with hierarchical recurrent networks[J]. arXiv preprint arXiv:1703.06345, 2017.
- [7] Ye Z, Ling Z H. Hybrid semi-Markov CRF for Neural Sequence Labeling[C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2018: 235-240.
- [8] Wu M, Liu F, Cohn T. Evaluating the Utility of Hand-crafted Features in Sequence Labelling[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 2850-2856.
- [9] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171-4186).
- [10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.
- [11] Akbik A, Blythe D, Vollgraf R. Contextual string embeddings for sequence labeling[C]//Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1638-1649.
- [12] Jia, C., Shi, Y., Yang, Q., & Zhang, Y. (2020, November). Entity enhanced BERT pre-training for Chinese NER. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 6384-6396).
- [13] Taher, E., Hoseini, S. A., & Shamsfard, M. (2019). Beheshti-NER: Persian named entity recognition Using BERT. In Proceedings of The First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019-Short Papers (pp. 37-42).