Enhancing Entity Boundary Detection for Better Chinese Named Entity Recognition

Chun Chen, Fang Kong*
School of Computer Science and Technology
Soochow University
20195227037@stu.suda.edu.cn, kongfang@suda.edu.cn

Abstract

In comparison with English, due to the lack of explicit word boundary and tenses information, Chinese Named Entity Recognition (NER) is much more challenging. In this paper, we propose a boundary enhanced approach for better Chinese NER. In particular, our approach enhances the boundary information from two perspectives. On one hand, we enhance the representation of the internal dependency of phrases by an additional Graph Attention Network(GAT) layer. On the other hand, taking the entity head-tail prediction (i.e., boundaries) as an auxiliary task, we propose an unified framework to learn the boundary information and recognize the NE jointly. Experiments on both the OntoNotes and the Weibo corpora show the effectiveness of our approach.

1 Introduction

Given a sentence, the NER task aims to identify the noun phrases having special meanings that predefined. Due to its importance on many downstream tasks, such as relation extraction(Ji et al., 2017), coreference resolution(Clark and Manning, 2016) and knowledge graphs(Zhang et al., 2019), NER has attracted much attention for long time.

In comparison with English, due to the lack of explicit word boundary and tenses information, Chinese NER is much more challenging. In fact, the performance of the current SOTAs in Chinese is far inferior to that in English, the gap is about 10% in F1-measure. In this paper, we propose a boundary enhancing approach for better Chinese NER.

Firstly, using Star-Transformer(Guo et al., 2019), we construct a lightweight baseline system. Benefit from the unique star topological structure, Star-Transformer is more dominant in representing long distance sequence, and thus, our baseline achieves comparable performance to the SOTAs. Considering the deficiency in the representation of local sequence information, we then try to enhance the local boundary information. In particular, our approach enhances the boundary information from two perspectives. On one hand, we add an additional GAT(Veličković et al., 2017) layer to capture the internal dependency of phrases. In this way, boundaries can be distinguished implicitly, while the semantic information within the phrase is enhanced. On the other hand, we add an auxiliary task to predict the head and tail of entities. In this way, using the framework of multi-tasking learning, we can learn the boundary information explicitly and help the NER task. Experiments show the effectiveness of our approach. It should be noted that, our approach obtains the new state-of-the-art results on both the OntoNotes and the Weibo corpora. That means our approach can perform well for both written and non-written texts.

2 Related Work

As is well known, most researches cast the NER task as a traditional sequence labelling problem, and many models extending the Bi-LSTM+CRF architecture are proposed (Huang et al., 2015; Chiu and Nichols, 2016; Dong et al., 2016; Lample et al., 2016; Ma and Hovy, 2016). Although the attention-based model, i.e., Transformer(Vaswani et al., 2017), has gradually surpassed the traditional RNN model(Zaremba et al., 2014) in various fields, Yan et al. (2019) has verified that the fully connected Transformer mechanism does not work well on NER. Until recently, some researches show that Star-Transformer can work well on NER owing to its lightweight topological structure(Guo et al., 2019; Chen et al., 2020). Moreover, lexical and dependent information has been widely used in this task (Zhang and Yang, 2018; Ma et al., 2020; Li et al., 2020; Gui et al., 2019; Sui et al., 2019; Tang et al., 2020) to better capture local semantic information.

In this paper, using Star-transformer as our base-

[*] Corresponding author.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Short Papers), pages 20–25
August 1–6, 2021. ©2021 Association for Computational Linguistics

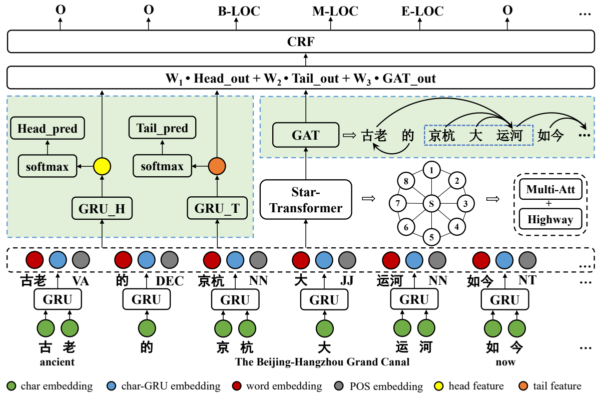

Figure 1: The general architecture for the boundary enhanced model.

line, we mainly focus on enhancing the boundary information to improve Chinese NER.

3 Model

We also treat NER as a sequence labeling task, decoding with a classical CRF(Lafferty et al., 2001). Figure 1 shows the complete model. We can find that the encoder of our model consists of three parts, i.e., GRU-based head and tail representation layer, Star-transformer based contextual embedding layer, and GAT-based dependency embedding layer.

3.1 Token embedding layer

Considering the lack of explit word boundary, we combine word-level represention with character, avoiding the error propagation caused by word segmentation.

For a given sentence, we represent each word and character by looking up the pre-trained word embeddings[1](Li et al., 2018). The sequence of character embeddings contained in a word will be fed to a bi-direction GRU layer. The hidden state of bi-direction GRU can be expressed as folowing:

$$\overrightarrow{h}_i^t = \overrightarrow{GRU}(x_i^t, \overrightarrow{h}_{i-1}^t) \quad (1)$$

$$\overleftarrow{h}_i^t = \overleftarrow{GRU}(x_i^t, \overleftarrow{h}_{i+1}^t) \quad (2)$$

$$h_i^t = [\overrightarrow{h}_i^t; \overleftarrow{h}_i^t] \quad (3)$$

where x_i^t is the token representation, $\overrightarrow{h}_i^t$ and $\overleftarrow{h}_i^t$ denote the t-th forward and backward hidden state of GRU layer.

The final token representation is obtained as equation(4) ∼ (6):

$$x_i^w = e(word_i) \quad (4)$$

$$x_i^c = GRU(e(char_i)) \quad (5)$$

$$x_i = [x_i^w; x_i^c; pos_i] \quad (6)$$

where [;] denotes concatenation, and pos_i is the Part-of-Speech tagging of $word_i$.

3.2 Star-transformer based contextual embedding layer

Star-Transformer abandons redundant connections and has an approximate ability to model the long-range dependencies. For NER task, entities are sparse, so it is unnecessary to pay attention on all nodes in the sentence all the time. We utilize this structured model to encode the words in a sentence, which shows comparable performance with the traditional RNN models, but with the capability of capturing long-range dependencies.

3.2.1 Multi-Head Attention

Transformer employs h attention heads to implement self-attention on an input sequence separately. The result of each attention head will be integrated together, called Multi-Head Attention.

Given a sequence of vectors X, we use a query vector Q to soft select the relevant information with attention:

$$Att(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}}) \cdot V \quad (7)$$

$$K = XW^K, V = XW^V \quad (8)$$

where W^K and W^V are learnable parameters. Then Multi-Head Attention can be defined as equation(9) ∼ (10):

$$MulAtt = (z_1 \oplus z_2 \oplus \cdots \oplus z_h) \cdot W^o \quad (9)$$

$$z_i = Att(QW_i^Q, KW_i^K, VW_i^V) \quad (10)$$

where $\oplus$ denotes concatenation, and W^o, W_i^Q, W_i^K, W_i^V are learnable parameters.

3.2.2 Star-Transformer Encoder

The topological structure of Star-Transformer is made up of one relay node and n satellite nodes. The state of i-th satellite node represents the feature of the i-th token in a text sequence. The relay node acts as a virtual hub to gather and scatter information from and to all the satellite nodes(Guo et al., 2019).

[1] https://github.com/Embedding/Chinese-Word-Vectors

Star-Transformer proposes a time-step cyclic updating method, in which each satellite node is initialized by the input vector, and the relay node is initialized as the average value of all tokens. The status of each satellite node is updated according to its adjacent nodes, including the previous node in the previous round h_{i-1}^{t-1}, the current node in the previous round h_i^{t-1}, the next node in the previous round h_{i+1}^{t-1}, the current node e^i and the relay node in the previous round s^{t-1}. The update process is shown in the equation(11) $\sim$ (12):

$$\begin{aligned} C_i^t &= [h_{i-1}^{t-1}; h_i^{t-1}; h_{i+1}^{t-1}; e^i; s^{t-1}] \quad (11) \\ h_i^t &= MulAtt(h_i^{t-1}, C_i^t, C_i^t) \quad (12) \end{aligned}$$

where C_i^t denotes contextual information of i-th.

The update of relay node is determined by the information of all the satellite nodes and the status of the previous round :

$$s^t = MulAtt(s^{t-1}, [s^{t-1}; H^t], [s^{t-1}; H^t]) \quad (13)$$

3.2.3 Highway Networks

Highway Networks(Srivastava et al., 2015) can alleviate the blocked gradient backflow when the network deepens. Such gating mechanisms can be of vital significance to Transformer(Chai et al., 2020). We use Highway Networks to mitigate the depth and complexity of Star-Transformer.

After calculating the Multi-Head Attention, a new branch dominated by Highway Networks joins in, indicating the self-updating and dynamic adjustment of satellite node.

$$\begin{aligned} g &= \sigma(w_1 h_i + b_1) \quad (14) \\ f(h_i) &= w_2 h_i + b_2 \quad (15) \\ HW(h_i) &= (1-g)\cdot h_i + g \cdot f(h_i) \quad (16) \end{aligned}$$

where w_1, w_2, b_1, b_2 are learnable parameters, and σ is the activation function.

Finally, the updated satellite node is denoted as:

$$h_i = HW(h_i) + MulAtt(h_i, C_i, C_i) \quad (17)$$

Highway Networks not only enhances the inherent characteristics of the satellite nodes, but also avoids gradient blocking.

3.3 GAT-based dependency embedding layer

In this work, we propose the use of dependencies between words to construct graph neural networks. The dependency is directional, and the current word is only related to the word with shared edge. This kind of directed linkage further obtains the internal structural information of the entity, enriching the sequential representation.

Graph Attention Networks(GAT)(Veličković et al., 2017), leveraging masked self-attention layers to assign different importance to neighbouring nodes, works well with our work.

The attention coefficient e_{ij} and α_{ij} represents the importance of node j to node i:

$$\begin{aligned} e_{ij} &= att(W\vec{h}_i, W\vec{h}_j) \quad (18) \\ \alpha_{ij} &= softmax_j(e_{ij}) \quad (19) \\ &= \frac{exp(e_{ij})}{\sum_{k\epsilon N_i} exp(e_{ik})} \quad (20) \\ &= \frac{exp(LeakyReLU(\vec{a}^T[Wh_i \oplus Wh_j]))}{\Sigma_{k\epsilon N_i} exp(LeakyReLU(\vec{a}^T[Wh_i \oplus Wh_k]))} \quad (21) \end{aligned}$$

A GAT operation with K independent attention heads can be expressed as:

$$\vec{h}'_i = \sigma(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\epsilon N_i}\alpha_{ij}^k W^k \vec{h}_j) \quad (22)$$

where $\oplus$ denotes concatenation, W and $\vec{a}$ are learnable parameters, N_i is the neighborhood of node i, σ is the activation function.

In addition to the strong focus on the associated nodes of GAT layer, it can well make up for the deficiency of Star-Transformer in capturing the internal dependency of the phrases.

3.4 GRU-based head and tail representation layer

While GAT is effective in capturing internal dependency within an entity, the boundary of the entity need to be strengthened. We then regard the entity boundary detection as binary classification task, which trains with NER at the same time, giving NER clear entity boundary information.

During training phase, two separate GRU layers are used to make head and tail prediction of the entities, whose hidden features are added with the output of GAT layer:

$$\begin{aligned} H_h &= GRU_{head}(x_i) \quad (23) \\ H_t &= GRU_{tail}(x_i) \quad (24) \\ H &= W_1 \cdot H_h + W_2 \cdot H_t + W_3 \cdot H_{GAT} \quad (25) \end{aligned}$$

W_1, W_2, W_3 are learnable parameters, and H is the final input for CRF.

	OntoNotes						Weibo						
	OntoNotes V4.0			OntoNotes V5.0			Named Entity			Nominal Mention			Overall
Models	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	F1(%)
Zhang and Yang (2018)	76.35	71.56	73.88	-	-	-	-	-	53.04	-	-	62.25	58.79
Ma et al. (2020)	77.31	73.85	75.54	-	-	-	-	-	56.99	-	-	61.41	61.24
Li et al. (2020)	-	-	76.45	-	-	-	-	-	-	-	-	-	63.42
Jie and Lu (2019)	-	-	-	77.40	77.41	77.40	-	-	-	-	-	-	-
Gui et al. (2019)	76.13	73.68	74.89	-	-	-	-	-	55.34	-	-	64.98	60.21
Sui et al. (2019)	75.06	74.52	74.79	-	-	-	67.31	48.61	56.45	75.15	62.63	68.32	63.09
Tang et al. (2020)	76.59	75.17	75.87	-	-	-	-	-	59.08	-	-	68.61	63.63
Star(baseline)	73.40	76.50	74.92	75.41	75.66	75.53	78.67	55.92	65.37	88.16	**69.07**	77.46	68.15
Star + GAT	77.33	76.03	76.67	77.03	79.90	78.44	77.30	59.72	67.38	**90.85**	66.49	76.79	68.34
Star + MultiTask	78.64	**80.78**	79.69	77.60	80.01	78.79	**80.39**	58.29	67.58	89.86	68.56	**77.78**	68.61
Star + GAT + MultiTask	**79.25**	80.66	**79.95**	**78.22**	**80.88**	**79.53**	78.92	**62.09**	**69.50**	88.67	68.56	77.33	**70.14**

Table 1: Performance on OntoNotes V4.0, OntoNotes V5.0 and Weibo. Named Entity is the same to the entity of OntoNotes, while Nominal Mention is the reference words which have the property of nouns.

3.5 Model Learning

Entities boundaries are not only the task we deal with, but the perfect natural assistance by NER, which transform from outside to inside of the mention and vice versa.

The multi-task loss function is composed of the categorical cross-entroy loss for boundary detection and entity categorical label prediction:

$$L_{multi} = L_{head} + L_{tail} + L_{label} \qquad (26)$$

4 Experiments

4.1 Datasets

The label in our work is marked by BIESO, and we use Precision(P), Recall(R) and $F1$ score($F1$) as evaluation metrics.

OntoNotes V4.0[2](Pradhan, 2011) is a Chinese dataset and consists of texts from news domain. We use the same split as Zhang and Yang (2018).

OntoNotes V5.0[3](Pradhan et al., 2013) is also a Chinese dataset from news domain, but with larger scale and more entity types. We use the same split as Jie and Lu (2019).

Weibo NER[4](Peng and Dredze, 2015) contains annotated NER messages drawn from the social meida Sina Weibo. We use the same split as Peng and Dredze (2015).

Additionally, the tool used to parse syntactic dependency in this paper is DDParser[5].

4.2 Results and Analysis

We conduct experiments on the OntoNotes and Weibo corpora and compare the results with the existing models, as shown in table 1[6].

	OntoNotes V4.0			OntoNotes V5.0		
error types	TE	UE	BE	TE	UE	BE
Star	2236	1912	151	1921	1896	139
Star + GAT	1787	1916	140	1877	1596	169
Star + MultiTask	1772	**1563**	114	1814	1590	127
Star + GAT + MultiTask	**1701**	1564	**108**	**1762**	**1505**	**121**

Table 2: Entity recognition errors of our models, including Type Error(TE), Unidentification Error(UE) and Boundary Error(BE).

We begin by establishing a Star-Transformer baseline, which is more effective on the smaller social media Weibo corpus than OntoNotes. Star-Transformer could be superior to all existing models in Weibo, at least 6.29%(F1) and 8.85%(F1) for Named Entity(NE) and Nominal Entity(NM).

Considering the structural peculiarity of OntoNotes, where entities have similar composition, we utilize GAT to simulate the feature inside the entity. The precision on the OntoNotes are both improved by 3.93% and 1.62%. Futhermore, boundary prediction used as multi-task has been trained with label classification, supplying local sequence information for NER. Tabel 2 shows the number of different entity recognition errors of our models, including Type Error(TE), Unidentification Error(UE) and Boundary Error(BE).The addition of entity head-tail prediction reduces the number of boundary errors on OntoNotes V4.0 by 37. There is no doubt that the boundary enhanced model are quite profitable to the recognition of both entity boundary and entity type.

For Weibo, NE and NM illustrate different performance. The more standard NE has a similar performance to OntoNotes, while NM shows less

[2] https://catalog.ldc.upenn.edu/LDC2011T13

[3] https://catalog.ldc.upenn.edu/LDC2013T19

[4] https://github.com/cchen-nlp/weiboNER

[5] https://github.com/baidu/DDParser

[6] Our code is available at: https://github.com/cchen-reese/Boundary-Enhanced-NER.

impact from GAT, due to its short length and non-structue.

Combining the respective advantages of the three layers above, an unified and lightweight model can be applied to Chinese NER, getting the new state-of-the-art results on both the OntoNotes and Weibo corpora.

5 Conclusion

In this paper, we mainly focus on the impact of boundary information on Chinese NER. We firstly propose a Star-transformer based NER system. Then both explicit head and tail boundary information and Dependency GAT-based implicit boundary information are combined to improve Chinese NER. Experiments on both the OntoNotes and the Weibo corpora show the effectiveness of our approach.

Acknowledgement

This work was supported by the National Key R&D Program of China under Grant No. 2020AAA0108600, Project 61876118 under the National Natural Science Foundation of China and the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

Yekun Chai, Jin Shuo, and Xinwen Hou. 2020. Highway transformer: Self-gating enhanced self-attentive networks. *arXiv preprint arXiv:2004.08178*.

Chun Chen, Mingyang Li, and Fang Kong. 2020. Lightweight named entity recognition for weibo based on word and character. In *Proceedings of the 19th Chinese National Conference on Computational Linguistics*, pages 402–413.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Kevin Clark and Christopher D Manning. 2016. Improving coreference resolution by learning entity-level distributed representations. *arXiv preprint arXiv:1606.01323*.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. Character-based lstm-crf with radical-level features for chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications*, pages 239–250. Springer.

Tao Gui, Yicheng Zou, Qi Zhang, Minlong Peng, Jinlan Fu, Zhongyu Wei, and Xuan-Jing Huang. 2019. A lexicon-based graph neural network for chinese ner. 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)*, pages 1039–1049.

Qipeng Guo, Xipeng Qiu, Pengfei Liu, Yunfan Shao, Xiangyang Xue, and Zheng Zhang. 2019. Star-transformer. *arXiv preprint arXiv:1902.09113*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

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*, volume 3060.

Zhanming Jie and Wei Lu. 2019. Dependency-guided lstm-crf for named entity recognition. *arXiv preprint arXiv:1909.10148*.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on chinese morphological and semantic relations. *arXiv preprint arXiv:1805.06504*.

Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. Flat: Chinese ner using flat-lattice transformer. *arXiv preprint arXiv:2004.11795*.

Ruotian Ma, Minlong Peng, Qi Zhang, Zhongyu Wei, and Xuan-Jing Huang. 2020. Simplify the usage of lexicon in chinese ner. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5951–5960.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Nanyun Peng and Mark Dredze. 2015. Named entity recognition for chinese social media with jointly trained embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 548–554.

Sameer Pradhan. 2011. Proceedings of the fifteenth conference on computational natural language learning: Shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway networks. *arXiv preprint arXiv:1505.00387*.

Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2019. Leverage lexical knowledge for chinese named entity recognition via collaborative graph network. 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)*, pages 3821–3831.

Zhuo Tang, Boyan Wan, and Li Yang. 2020. Word-character graph convolution network for chinese named entity recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: Adapting transformer encoder for name entity recognition. *arXiv preprint arXiv:1911.04474*.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Wen Zhang, Bibek Paudel, Wei Zhang, Abraham Bernstein, and Huajun Chen. 2019. Interaction embeddings for prediction and explanation in knowledge graphs. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 96–104.

Yue Zhang and Jie Yang. 2018. Chinese ner using lattice lstm. *arXiv preprint arXiv:1805.02023*.