A Progressive Convolutional Network-Based Method for Agricultural Named Entity Recognition (Postprint)

Authors: Ji Jie, Jinzhou, Rujing Wang, Liu Haiyan, Li Zhiyuan

Date: 2023-05-15T00:00:00+00:00

Abstract

Currently, research on named entity recognition based on Pre-trained Language Models (PLM) , when confronted with issues in the agricultural domain such as complex entity naming conventions and ambiguous entity boundaries, exclusively utilizes the output representation from the final layer of PLM and invariably introduces knowledge or operations from external sources to enhance entity representations, thereby neglecting the rich, multi-level linguistic information inherently present across the internal layers of the model. To address these challenges, we propose a named entity recognition method based on a progressive convolutional network. This approach first stores the output representations from each layer obtained after processing natural sentences through PLM; subsequently, it employs progressive convolution as a feature extraction mechanism for comprehensive layer-wise information, performing sequential convolutions on the stored intermediate layer outputs. The model emphasizes full-layer information, including the previously overlooked shallow-layer outputs, as existing research indicates that sentence embeddings from layers closer to the input contain more coarse-grained information such as phrases and word groups. For agricultural named entity recognition with ambiguous boundaries, the critical phrase delimitation information may be implicitly encoded in these neglected shallow embeddings, thereby providing valuable assistance for addressing named entity recognition challenges in the agricultural domain. Without requiring external information, the method fully leverages the results obtained from the already expended computational resources to enhance sentence representation embeddings; finally, a globally optimal sequence is generated through a Conditional Random Field (CRF) model. On a constructed agricultural dataset encompassing four categories of agricultural entities—crop varieties, diseases, pests, and pesticides—the proposed method achieves a 3.61% improvement in the comprehensive F1 score compared to the Bidirectional Encoder Representation from Transformers (BERT) model, and also demonstrates strong perfor-

mance on public datasets, with the F1 score reaching 94.96% on the MSRA dataset. These results demonstrate that the progressive convolutional network can enhance the model's representation capability for natural language and offers advantages in named entity recognition tasks.

Full Text

Preamble

Progressive Convolutional Net Based Method for Agricultural Named Entity Recognition

JI Jie1,2, JIN Zhou1, WANG Rujing1,2*, LIU Haiyan1,2, LI Zhiyuan1,2
(1. Institute of Intelligent Machinery, Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China;
2. University of Science and Technology of China, Hefei 230026, China)

Abstract: Current research on named entity recognition (NER) based on pre-trained language models (PLMs) faces challenges in the agricultural domain, such as complex entity naming conventions and ambiguous entity boundaries. Existing methods only utilize the final layer's representation output and rely on external knowledge or operations to enhance entity representations, neglecting the rich hierarchical linguistic information inherently present within the model's internal layers. To address these issues, this study proposes a NER method based on progressive convolutional networks. The method first stores the output representations from each layer obtained when natural sentences pass through the PLM. It then employs progressive convolution as a feature extraction mechanism for full-layer information, sequentially convolving the stored intermediate layer outputs. This approach emphasizes complete layer information, including previously ignored shallow outputs. Research indicates that sentence embeddings from layers closer to the input contain more coarse-grained information such as phrases and word groups, which may provide crucial boundary information for ambiguous agricultural named entities. Without requiring external information, the method fully leverages the computational results already obtained to enhance sentence representation embeddings. Finally, a Conditional Random Field (CRF) model generates globally optimal sequences. On a constructed agricultural dataset containing four categories of agricultural entities (crop varieties, diseases, pests, and pesticides), the proposed method achieves a comprehensive F1-score improvement of 3.61 percentage points compared to the BERT (Bidirectional Encoder Representation from Transformers) baseline. It also demonstrates strong performance on public datasets, achieving 94.96% F1 on the MSRA dataset, indicating that progressive convolutional networks can enhance model representation capabilities for natural language and offer advantages in NER tasks.

Keywords: agricultural named entity recognition; pre-trained language model; convolutional network; representation aggregation; deep learning

1 Introduction

With the advancement of agricultural informatization, agricultural data on the internet has grown exponentially. Utilizing this data enables research on agricultural knowledge services such as question answering and knowledge graph construction. Identifying meaningful nouns or phrases from unstructured agricultural texts and classifying them—such as crop varieties, diseases, pests, and pesticide names—constitutes agricultural named entity recognition. This critical step supports subsequent acquisition of high-quality semantic knowledge and enables agricultural information extraction and semantic retrieval. Improving NER accuracy effectively facilitates knowledge services in agriculture, allowing decision-makers, producers, and researchers to obtain comprehensive and highly relevant information promptly and accurately, thereby enhancing overall agricultural development.

The emergence of pre-trained models has brought new opportunities to natural language processing [1,2]. Currently effective NER methods are implemented based on pre-trained language models. Specifically, the BERT (Bidirectional Encoder Representation from Transformers) model [3] has been widely applied to NER in both open and vertical domains in recent years. BERT is a deep model stacked with multiple Transformer [4] layers with the ability to encode contextual information. These language models use attention mechanisms to learn contextual information, converting natural language into semantically rich sentence embeddings. Leveraging this characteristic, Yang and Dong [5] proposed using BERT to optimize the traditional BiGRU-CRF (Bidirectional Gating Recurrent Unit + Conditional Random Field) method for Chinese NER. Gan et al. [6] combined BERT with BiLSTM-CRF, while Gao et al. [7] applied this framework to the CCKS2020 electronic medical record dataset. Chang et al. [8] simultaneously used BiLSTM and IDCNN (Iterated Dilated Convolutional Neural Networks) for feature extraction from sentence embeddings, then fused both features to obtain stronger representations. Li et al. [9] introduced external entity knowledge into BERT input, expanding traditional lattice structures into flat structures.

However, the agricultural domain presents challenges including complex entity name composition, long entity names [13], ambiguous word boundaries, and low recognition rates for rare words [14]. Existing NER methods focus on introducing external knowledge or operations to enhance sentence features, increasing downstream model complexity for accuracy gains while performing unnecessary operations on enhanced information. Simultaneously, aligning externally introduced data reduces model generalizability. Research shows [2,15,16] that each BERT layer learns different dimensions of linguistic information: Transformer layers closer to the input produce sentence embeddings containing more phrase-level and syntactic information, while deeper encoding layers focus more on semantic information. For agricultural NER, coarser-grained information such

as phrases and syntax may be more useful [17]. Using only the final layer representation may overlook shallow information beneficial for NER. For instance, in agricultural knowledge graph question answering, user queries are often brief with typical sentence components. In the query "Which crops does wheat powdery mildew affect?", both "wheat" and "powdery mildew" can be independent entities, but the complete entity "wheat powdery mildew" is preferred. Therefore, leveraging coarse-grained linguistic information better captures long agricultural entity names and plays a crucial role. The key phrase boundary information for ambiguous agricultural NER may be 隐含 in these ignored shallow embeddings.

To address these agricultural NER challenges and fully mine the different dimensions of linguistic information within PLMs, this study proposes a NER method combining PLMs with progressive convolutional networks. This approach aggregates all encoder layer outputs from the PLM using progressive convolution operations, extracting different dimensions of linguistic information while emphasizing both shallow and deep layer information. This enables the model to possess deep semantic information while integrating coarse-grained information beneficial for agricultural entity names, enhancing sentence entity representation capabilities. Without adding external information, it fully utilizes computational results already obtained to enhance natural language representation embeddings. Experimental results demonstrate the proposed method's effectiveness across multiple PLMs, improving accuracy on various NER datasets.

2 Methodology

2.1 Representation Layer

The representation layer encodes natural language sequences into vector representations. Using a PLM to encode input sentences yields a set of context-aware embedding representations. Taking BERT as an example, the representation layer structure is shown in Figure 2 [Figure 2: see original paper], where Trm represents the Encoder part of Transformer [11].

Given a PLM, the input sequence is encoded. Since the PLM has a multi-layer Transformer structure, we obtain a sentence representation set LS as shown in formula (1):

$$LS = PLM(S) = \{L_1, L_2, ..., L_l\}$$

where $L_i \in \mathbb{R}^{n \times h}, i \in \{1, 2, ..., l\}$ represents the sentence representation encoded by the i-th layer of the PLM; l is the depth of the PLM; $\mathbb{R}$ denotes the set of real matrices; n and h represent sentence length and PLM hidden dimension, respectively.

Different layers of the PLM encode representations with different focuses, so the obtained representation set contains multiple dimensions of linguistic infor-

mation such as phrases, lexical features, word order, and sentence semantics. Therefore, the representation set LS can more fully represent the input sentence.

2.2 Progressive Convolutional Network Construction

Unlike existing methods, this study designs a progressive convolutional network with depth $l-1$, where each layer has identical structure as shown in Figure 3 [Figure 3: see original paper].

Each network layer consists of three components: layer concatenation, convolutional layer, and normalization. Layer concatenation combines the previous layer's fused representation $AR_{i-1} \in \mathbb{R}^{n\times h}$ with the current layer's sentence embedding $L_i \in \mathbb{R}^{n\times h}$ to obtain a multi-dimensional mixed representation $MR_i \in \mathbb{R}^{2\times n\times h}, i \in \{1,2,...,l\}$. This concatenation facilitates subsequent convolution operations on both, as shown in formula (2):

$$MR_i = \text{concat}(AR_{i-1}, L_i) = [AR_{i-1}; L_i]$$

The convolutional layer fuses the previous layer's output AR_{i-1} with the current layer's embedding L_i. For the convolutional layer at level c of the progressive network, with input MR_c, convolution kernel $R^{2\times w\times b\times 1}$ (where w and b are kernel length and width), and output $R^{w\times b}$, E_c is computed as shown in formula (3):

$$E_c = MR_c \otimes k_c$$

where $E_c(x,y) = \sum_{i,j}(AR_{x+i,y+j} \cdot k_{c,0,i,j} + LR_{x+i,y+j} \cdot k_{c,1,i,j})$, $LR_{x,y}$ is the element at row x, column y of the current layer's sentence representation matrix, and $AR_{x,y}$ is the element at row x, column y of the AR matrix.

Through layer connection and convolution, the sentence embedding dimensions remain unchanged before and after fusion. Compared to before fusion, the post-fusion embedding extracts features from the current layer. For a specific position in the sequence, convolution enables it to learn contextual representations, which benefits NER tasks as learning representations of other characters in an entity helps recognition.

The normalization layer ensures the convolved sentence embedding maintains consistent magnitude with pre-convolution embeddings, facilitating next-layer fusion. It also introduces nonlinear transformations, improving network expressiveness. Unlike computer vision's Batch Normalization which normalizes the same feature across different samples in a batch, this method uses Layer Normalization, which normalizes different features within a single sample. This preserves comparability among semantic vectors of different words in the same sentence context, making it more suitable for NLP tasks and helping prevent overfitting. As shown in Figure 3, given E_c, its normalized value AR_c is calculated using formulas (4)-(6):

$$\mu_i = \frac{1}{n}\sum_{j=1}^{n} E_{c_{j,i}}$$

$$\sigma_i = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(E_{c_{j,i}} - \mu_i)^2}$$

$$AR_{c_{j,i}} = g_i \cdot \frac{E_{c_{j,i}} - \mu_i}{\sigma_i} + b_i$$

where μ_i is the mean of column i in E_c; n is sentence length; σ_i is the variance of column i; g_i and b are trainable parameters. This normalization is applied on the same hidden dimension of sentence embeddings.

2.3 Decoding Layer

The decoding layer converts the aggregated distributed representation AR_c from Section 2.2 into corresponding label sequences. To obtain optimal label sequences, CRF [22] is used for decoding. For an input sentence sequence $S = \{s_1, s_2, ..., s_n\}$ with corresponding label sequence $y = \{y_1, y_2, ..., y_n\}$ where $y_i \in Y$ and Y is the set of entity categories, the conditional probability is calculated using formula (7):

$$p(y|S) = \frac{\prod_{i=1}^{n} Q_{i,y_i} \cdot e^{\sum_{i=1}^{n-1} A_{y_i,y_{i+1}}}}{\sum_{y^* \in Y^*} \prod_{i=1}^{n} Q_{i,y_i^*} \cdot e^{\sum_{i=1}^{n-1} A_{y_i^*,y_{i+1}^*}}}$$

where Q is the probability matrix, Q_{i,y_i} represents the probability of position i being labeled as y_i; A is the state transition matrix, $A_{y_i,y_{i+1}}$ represents the probability of transitioning from label y_i to y_{i+1}; Y^* is the set of all possible label sequences for input sequence S. Initial Q is obtained through a fully connected layer in the CRF, while A is randomly initialized by the CRF model. The probability matrix Q and transition matrix A jointly determine label sequence selection, with the highest-scoring path chosen as the final result.

2.4 Training Phase

In formula (7), Q and A are learnable parameters obtained through model training. The loss function during training uses the log-likelihood of conditional probability, as shown in formula (8):

$$L(S, y) = -\log(p(y|S))$$

During backpropagation, Q and A are continuously adjusted based on the loss function, ultimately generating transition matrix A that constrains label sequential relationships. This study selects the Adam optimization algorithm for learning model parameters.

After learning matrices Q and A, during prediction, the Viterbi algorithm solves for the label sequence y^* that maximizes conditional probability $p(y|S)$, as shown in formula (9):

$$y^* = \arg\max_y p(y|S)$$

3 Model Validation and Experimental Design

To validate the proposed method's effectiveness, BERT [3], NEZHA [23], and BERT-wwm [24] were selected as model representation layers. These models are all base models with 12-layer Transformer [11] structures. Input sequences pass through these PLMs for contextual learning to obtain corresponding representation layer sets, then the proposed progressive convolutional network module is added between the PLM and CRF decoding layer to obtain enhanced sentence aggregation representations for subsequent NER.

3.1 Data Acquisition and Evaluation Metrics

Public Datasets: The People's Daily (PeopleDaily) NER dataset and Microsoft Research Asia (MSRA) NER dataset were selected. Both datasets categorize named entities into person names, locations, and organization names. Dataset statistics are shown in Table 1 .

Agricultural Dataset: Publicly annotated datasets in the agricultural domain are scarce [14]. Therefore, the AgriNER dataset was constructed by manually collecting and organizing data from existing agricultural product ontology knowledge bases. Concepts in ontologies abstract instances, pointing to categories of instances with similar properties. Consequently, this study treats all concepts and instances in the ontology as named entities. Information is detailed in Table 2 and Table 3 . The task identifies agricultural product named entities and classifies them into five categories: Product Class (PC), Product Instance (PI), Disease and Pest Class (DPC), Disease and Pest Instance (DPI), and Region (RI).

The BIO tagging scheme was adopted, where B marks entity beginnings, I marks entity continuations, and O marks non-entity parts.

Evaluation metrics include precision (P), recall (R), and F1-score (harmonic mean of P and R) to assess model performance on NER tasks.

3.2 Experimental Environment and Parameter Settings

The experimental environment is detailed in Table 4 . Using BERT, NEZHA, and BERT-wwm as representation layers, and since the constructed agricultural corpus is smaller than public datasets, models were first tested on public datasets to determine hyperparameters including convolution kernel size and PLM selection.

Hyperparameters were determined through multiple experiments, as shown in Table 5 . Additionally, training epochs varied based on representation layer selection: except when using BERT on MSRA dataset (epoch=3), all other experiments used epoch=5.

4 Experimental Results Analysis and Discussion

All experiments were repeated three times, with averages taken to neutralize effects of random parameter initialization.

4.1 Public Dataset Experimental Results Analysis

4.1.1 Impact Analysis on Different Representation Layers Results in Tables 6-8 show the proposed progressive convolutional network-based NER method achieves higher F1-scores on both PeopleDaily and MSRA datasets compared to other models. BERT-BiLSTM [7] serves as a baseline combining BERT with deep neural networks, while Sesame [19] and JAM [18] represent alternative multi-layer aggregation models.

As shown in Table 6, the proposed method outperforms Sesame and JAM multi-layer representation fusion models. Compared to BERT, the method achieves F1 improvements of 0.51% on PeopleDaily and 0.84% on MSRA. Results demonstrate the method enhances natural language representation capability and improves NER accuracy.

Tables 7 and 8 show experimental results based on NEZHA and BERT-wwm models. The progressive convolutional network fusion method shows significant effects on both models: F1 improves by 0.19% on PeopleDaily and 0.23% on MSRA with NEZHA; and improves by 0.24% on PeopleDaily and 0.53% on MSRA with BERT-wwm. The method proves effective not only for BERT but also for PLMs with identical structures.

The performance gain is most pronounced on BERT, less so on BERT-wwm and NEZHA (Figure 4 [Figure 4: see original paper]). This difference stems from model characteristics: BERT-wwm and NEZHA improve upon BERT using whole-word masking and relative position encoding in attention matrices. From an information theory perspective, these optimizations enhance BERT's encoding capability, reducing uncertainty in the encoding process and decreasing the amount of information that representation aggregation can enhance. Consequently, the three models exhibit hierarchical effectiveness improvements through progressive convolutional network aggregation.

4.1.2 Comparison with Different Fusion Methods Sesame and JAM models perform worse than the original model. JAM simply linearly combines decoder layer representations, primarily using gating networks to regulate information flow to the next layer, requiring self-learned weights. Sesame uses squeeze-and-excitation operations to obtain weight factors for BERT layers, weighting full-layer outputs accordingly. Both approaches learn weights based on BERT layers' performance on downstream tasks, strengthening more adaptive intermediate representations while weakening less important ones. However, strengthening or weakening an entire intermediate layer's distributed representation still ignores some information. The proposed method preserves full-layer information on average, relatively limiting model learning freedom and forcing attention to each layer's information, better aggregating PLM embeddings and mining linguistic properties learned from large-scale corpora.

4.1.3 Impact of Convolution Operations and Kernel Size To explore convolution layer effects, experiments were conducted with BERT as representation layer using a 5$×$768 kernel size (768 being BERT's hidden dimension), commonly effective in classification tasks. Results in Table 9 show parameter increases over BERT's original 110M parameters and per-epoch training time (Te).

Results indicate smaller kernels are more suitable. First, large kernels show no F1 improvement, indicating ineffective representation aggregation. Second, larger kernels increase parameters and training time. The progressive convolutional network for NER tasks is better suited to small kernels. Compared to BERT, the proposed model shows no significant time or space overhead increase, keeping complexity manageable.

4.2 Agricultural Product Dataset Experiments

Based on Section 4.1's analysis showing BERT as the most effective baseline and kernel size significantly impacting performance, agricultural NER experiments used BERT with different kernel sizes. Results in Table 10 show the method significantly improves R and F1 compared to traditional methods. When kernel size is 5$×$5, $F1improvementismaximal; at3$×$3$, F1 is minimal but still outperforms traditional methods. R shows similar patterns on public datasets. Larger kernels decrease P, but the reduction is smaller than R's improvement.

4.2.2 Performance Analysis on Different Entity Categories To analyze model performance across entity categories, evaluation metrics were computed per category on AgriNER. Since R and P patterns mirror F1, F1 values are reported in Table 11 . Traditional BERT shows large F1 fluctuations across categories (53.73-98.69), highest for Region entities and lowest for Product Class. With 5$×$5 kernels, the proposed method shows minimal F1 fluctuation (71.60-98.87), also highest for Region and lowest for Product Class. All models perform worst on Product Class entities due to high similarity between Product

Class and Product Instance entities (e.g., “pea category” includes instances like “pea,” “fresh pea,” “snow pea”). Similar patterns occur for Disease/Pest Class vs. Instance.

4.2.3 Analysis of Labeled Data Impact on Evaluation Analyzing precision formula $P = TP/(TP + FP)$ reveals TP and FP statistics. Compared to BERT, the proposed model shows minimal TP differences (<10 instances, ~0.7% of average TP) but higher FP counts (>10 instances, ~25% of average FP), causing lower precision values. Manual examination of false positives reveals some detected entities are valid natural language entities absent from test set labels. The agricultural knowledge base annotation process involved random character replacement, introducing noise. Similar patterns occur in non-nested public datasets like MSRA. The model may over-integrate shallow information, making entity recognition overly sensitive and failing to match label set entities. This underscores the importance of labeling quality and strategy for model evaluation.

5 Conclusion

This study addresses the problem of existing PLM-based NER methods that neglect rich hierarchical information within PLM layers, using only final layer outputs and underutilizing the model. A progressive convolutional network is proposed to aggregate all encoder layer outputs. The method uses progressive convolution to extract different dimensions of linguistic information, emphasizing both shallow and deep layer information. This enables the model to possess deep semantic information while integrating coarse-grained information beneficial for agricultural entity names, enhancing sentence entity representation. Compared to BERT, the method achieves F1 improvements of 0.51% on PeopleDaily, 0.84% on MSRA, and 3.61% on AgriNER. Results demonstrate effectiveness on both public datasets and agricultural applications, better locating entity positions and addressing fuzzy boundaries and low recognition rates for specialized terms. The effectiveness of small kernels is validated, confirming that progressive networks enhancing representations improve NER accuracy.

However, for non-nested entity name lengths, the model integrates excessive shallow information, leaving room for improvement in capturing contextual semantic information. Since semantic information also serves as important reference for entity name length cutoff points, future research should explore enhancing deep information while maintaining shallow information fusion.

Conflict of Interest Statement: This study has no conflicts of interest among researchers or with publicly available research findings.

References

[1] QIU X P, SUN T X, XU Y G, et al. Pre-trained models for natural language processing: A survey[J]. Science China technological sciences, 2020, 63(10):

1872-1897.

[2] SEVASTJANOVA R, KALOULI A, BECK C, et al. Explaining contextualization in language models using visual analytics[C]// 2021 59th Association for Computational Linguistics (ACL). Stroudsburg, PA, USA: Association for Computational Linguistics, 2021: 464-476.

[3] DEVLIN J, CHANG M-W, LEE K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]// North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT). Stroudsburg, PA, USA: Association for Computational Linguistics, 2019: 4171-4186.

[4] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// 2017 31st Annual Conference on Neural Information Processing Systems (NIPS). La Jolla, California, USA: Neural Information Processing Systems, 2017: 6000-6100.

[5] YANG P, DONG W Y. Chinese named entity recognition method based on BERT embedding[J]. Computer engineering, 2020, 46(4): 40-45, 52.

[6] GAN Y, YANG R S, ZHANG C F, et al. Chinese named entity recognition based on BERT-transformer-BiLSTM-CRF model[C]// 2021 7th International Symposium on System and Software Reliability (ISSSR). Piscataway, NJ, USA: IEEE, 2021: 109-118.

[7] GAO W C, ZHENG X H, ZHAO S S. Named entity recognition method of Chinese EMR based on BERT-BiLSTM-CRF[J]. Journal of physics. Conference series, 2021, 1848(1): ID 012083.

[8] CHANG Y, KONG L, JIA K J, et al. Chinese named entity recognition method based on BERT[C]// 2021 IEEE International Conference on Data Science and Computer Application (ICDSCA). Piscataway, NJ, USA: IEEE, 2021: 537-540.

[9] LI X, YAN H, QIU X, et al. FLAT: Chinese NER Using Flat-Lattice Transformer; proceedings of the ACL, F, 2020[C]// 2020 58th Annual Meeting of the Association for Computational Linguistics (ACL). Stroudsburg, PA, USA: Association for Computational Linguistics, 2020: 6836-6842.

[10] JU S G, LI T N, SUN J P. Chinese fine-grained name entity recognition based on associated memory networks[J]. Journal of software, 2021, 32(8): 2545-2556.

[11] WANG X Y, JIANG Y, BACH N, et al. Improving named entity recognition by external context retrieving and cooperative learning[J/OL]. arXiv: 2105.03654, 2021.

[12] NIE Y Y, TIAN Y H, SONG Y, et al. Improving named entity recognition with attentive ensemble of syntactic information[C]// Findings of the Association for Computational Linguistics: EMNLP 2020. Stroudsburg, PA, USA:

Association for Computational Linguistics, 2020: 4231-4245.

[13] LI L, ZHOU H, GUO X C, et al. Named entity recognition of diseases and insect pests based on multi source information fusion[J]. Transactions of the Chinese society for agricultural machinery, 2021, 52(12): 253-263.

[14] ZHAO P F, ZHAO C J, WU H R, et al. Named entity recognition of Chinese agricultural text based on attention mechanism[J]. Transactions of the Chinese society for agricultural machinery, 2021, 52(1): 185-192.

[15] JAWAHAR G, SAGOT B, SEDDAH D. What does BERT learn about the structure of language?[C]// 2019 57th Annual Meeting of the Association for Computational Linguistics (ACL). Stroudsburg, PA, USA: Association for Computational Linguistics, 2019: 3651-3657.

[16] ROGERS A, KOVALEVA O, RUMSHISKY A. A primer in BERTology: What we know about how BERT works[J]. Transactions of the association for computational linguistics, 2020, 8: 842-866.

[17] JIE Z M, LU W. Dependency-guided LSTM-CRF for named entity recognition[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Stroudsburg, PA, USA: Association for Computational Linguistics, 2019: 4231-4245.

[18] ZHANG Z B, WU S, JIANG D W, et al. BERT-JAM: Maximizing the utilization of BERT for neural machine translation[J]. Neurocomputing, 2021, 460: 84-94.

[19] SU T C, CHENG H C. SesameBERT: Attention for anywhere[C]// 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA). Piscataway, NJ, USA: IEEE, 2020: 363-369.

[20] HU J, SHEN L, ALBANIE S, et al. Squeeze-and-excitation networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2020, 42(8): 2011-2023.

[21] JIANG Z, YU W, ZHOU D, et al. ConvBERT: Improving BERT with Span-based Dynamic Convolution[J/OL]. arXiv:2008.02496 [cs.CL], 2020.

[22] SUN C J, GUAN Y, WANG X L, et al. Rich features based Conditional Random Fields for biological named entities recognition[J]. Computers in biology and medicine, 2007, 37(9): 1327-1333.

[23] WEI J, REN X, LI X, et al. NEZHA: Neural contextualized representation for Chinese language understanding[J/OL]. arXiv:1909.00204v3 [cs.CL], 2009.

[24] CUI Y M, CHE W X, LIU T, et al. Pre-training with whole word masking for Chinese BERT[J]. IEEE/ACM transactions on audio, speech and language processing, 2021, 29: 3504-3514.

Note: Figure translations are in progress. See original paper for figures.