Attention-based Multi-level Feature Fusion for Named Entity Recognition

Zhiwei Yang[1,5] , **Hechang Chen**[2,5*] , **Jiawei Zhang**[3] , **Jing Ma**[4] and **Yi Chang**[2,5,6]
[1]College of Computer Science and Technology, Jilin University, Changchun, China
[2]School of Artificial Intelligence, Jilin University, Changchun, China
[3]IFM Lab, Department of Computer Science, Florida State University, Tallahassee FL, USA
[4]Department of Computer Science, Hong Kong Baptist University, Hong Kong, China
[5]Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, China
[6]International Center of Future Science, Jilin University, Changchun, China
yangzw18@mails.jlu.edu.cn, chenhc@jlu.edu.cn, jiawei@ifmlab.org, mmjjblue@gmail.com,
yichang@jlu.edu.cn

Abstract

Named entity recognition (NER) is a fundamental task in the natural language processing (NLP) area. Recently, representation learning methods (e.g., character embedding and word embedding) have achieved promising recognition results. However, existing models only consider partial features derived from words or characters while failing to integrate semantic and syntactic information (e.g., capitalization, inter-word relations, keywords, lexical phrases, etc.) from multi-level perspectives. Intuitively, multi-level features can be helpful when recognizing named entities from complex sentences. In this study, we propose a novel framework called attention-based multi-level feature fusion (AMFF), which is used to capture the multi-level features from different perspectives to improve NER. Our model consists of four components to respectively capture the local character-level, global character-level, local word-level, and global word-level features, which are then fed into a BiLSTM-CRF network for the final sequence labeling. Extensive experimental results on four benchmark datasets show that our proposed model outperforms a set of state-of-the-art baselines.

1 Introduction

Named entity recognition aims to identify entities that have similar characteristics from raw text and assign them identical tags such as PER (Person), ORG (Organization), LOC (Location), etc. As a fundamental task of information extraction, NER has been studied for various tasks, including part-of-speech (POS) tagging, chunking, and semantic role labeling (SRL) [Collobert *et al.*, 2011]. Considering the diversity and complexity of natural language, named entities generally have multi-features from different perspectives: character-level and word-level features from local and global perspectives, as illustrated in Figure 1. We can see that the named entity associated with polysemous words (Washington) should take the capitalization 'W', keyword 'in', and lexical phrase into consideration.

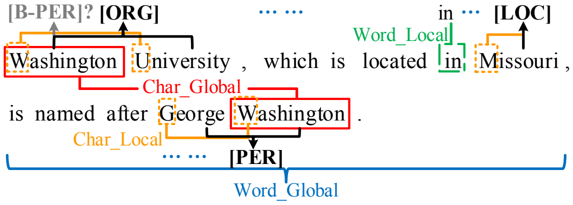
[B-PER]?
[ORG]
··· ···
in ···
[LOC]
Word_Local
Washington University , which is located in Missouri ,
Char_Global
is named after George Washington .
Char_Local
··· ···
[PER]
Word_Global

Figure 1: A brief illustration of the multi-level features. There are four kinds of character-level and word-level features from local and global perspectives: Char_Local, Char_Global, Word_Local, Word_Global, such as the capitalization (orange), the polysemous word 'Washington' (red), the keyword 'in' (green), and the lexical phrase (blue) which frequently occur together. The output named entities are shown in black.

Previous knowledge-based approaches for NER merely depend on handcrafted rules and domain-specific dictionaries to recognize named entities [Rahem and Omar, 2014; Quimbaya *et al.*, 2016]. However, such an endeavor is manual and is thus prone to poor coverage. Traditional approaches use supervised machine learning algorithms that incorporate a wide variety of hand-crafted features. To alleviate the heavy manual effort associated with these approaches, neural models are proposed to learn the implicit features by utilizing word-level embedding [Huang *et al.*, 2015], character-level embedding [Kuru *et al.*, 2016] or both [Akbik *et al.*, 2018]. However, these methods largely ignore or oversimplify the correlations among the features learned from different perspectives such as word-level and character-level. Although existing NER approaches based on the combination of word-level embeddings and character-level embeddings have achieved competitive results [Xin *et al.*, 2018; Akbik *et al.*, 2018; Beltagy *et al.*, 2019], they pay little attention to fusing multi-features, which may lead to information omission. Taking the sentence in Figure 1 as an example, the polysemous word 'Washington' of 'Washington University' (*ORG*) might be mislabelled as *B-PER* without Char_Global features by merely taking adjacent words or character features in context into consideration, which may result in 'Washing-

[*] Corresponding author: chenhc@jlu.edu.cn

ton University' being classified as *PER* by mistake.

To this end, we propose an attention-based multi-level feature fusion framework for NER, where the multi-level semantic and syntactic features of a given input sequence can be simultaneously captured from different views, as shown in Figure 1. Inspired by the transformer network[Vaswani *et al.*, 2017], we explicitly employ four components for different purposes, namely the local character-level component, global character-level component, local word-level component, and global word-level component to process the input sequences. Then the fusion representation from the multi-level features is fed into the BiLSTM-CRF network for the final prediction.

The main contributions of this work are summarized as follows:

- An attention-based multi-level feature fusion (AMFF) framework is proposed for NER, which enables the multi-level features from diverse word-level and character-level perspectives to be integrated. To the best of our knowledge, this is the first effort in the literature that uses attention mechanisms to capture multi-level features of text from different perspectives.
- In this framework, local character-level, global character-level, local word-level, and global word-level components are designed to capture the features from the perspectives of capitalization, inter-word relations, keywords, and lexical phrases, respectively. It simplifies the problem and improves interpretability.
- Extensive validations on real-world benchmark datasets compared with the state-of-the-art models demonstrate the superiority of our proposed AMFF. Systematic analyses show an in-depth understanding of each component and the robustness of the AMFF framework.

2 Related Work

Existing NER studies are primarily based on word embeddings, character embeddings, and other embedding combinations, which are summarized as follows.

The first word embedding-based NER approach [Collobert *et al.*, 2011] adopted convolution neural networks (CNNs) to produce local features, and a conditional random field (CRF) layer to predict entity attributes, which achieved better performance than previous studies. In order to address the problem of long-distance dependency, the CNN layer was replaced by the bi-directional long short-term memory (BiLSTM) layer for the better selection of the global features [Huang *et al.*, 2015]. Furthermore, the combination of both CNN and BiLSTM highlights the performance improvement for linguistic sequence labeling [Liu *et al.*, 2018]. However, these methods ignore the effect at the character level, which may lead to the omission of important information.

Word representations can be obtained from character-level embeddings as each sentence can be regarded as a character sequence. A character-level convolutional neural network (CharCNN) [Kim *et al.*, 2016] is proposed to extract local sub-word information, relying on LSTM to deal with context features and the softmax function for the final prediction, which highlights the character embeddings for NER. Moreover, character-level recurrent neural networks (CharRNN) [Kuru *et al.*, 2016] choose global features from the context and boost the performance of NER by using CRF. In addition, it achieves better performance in handling multiple languages [Dong *et al.*, 2016]. However, methods based on character embeddings do not attach importance to word-level features, which may be biased as well.

Recent advances in NER demonstrate its great advantage in recognizing named entities based on the combination of embeddings [Akbik *et al.*, 2018; Yoon *et al.*, 2019]. In order to capture both global and local features, existing methods rely on more types of embeddings, for example, BERT-based methods [Beltagy *et al.*, 2019] taking token embeddings, segment embeddings, and position embeddings into account for NER. Moreover, other additional information such as affix embeddings can also be used to augment the NER architecture [Yadav *et al.*, 2018]. In addition, the multitask learning strategy [Zheng *et al.*, 2019] that divides the original task into multiple subtasks offers a new approach for nested NER. Although these methods achieve competitive results, few of them explore the attention mechanism for multi-level feature selection in NER, which does not make full use of the complete information.

Therefore, different from the existing models that mine information from merely one aspect, our model focuses on leveraging multi-level features from different perspectives, which can obtain more types of information and make a comprehensive final prediction.

3 The Proposed Framework

In this section, we introduce a novel method called AMFF, which can obtain multi-level features by parallel components and fuse them for sequence labeling.

Given an input sentence S composed of a sequence of words $w_1 w_2 \cdots w_n$, where n is the total number of words, we assign each word w_i with one label y_i that takes one possible class from the named entity label set: $\mathbf{y} = \{B\text{-}ORG, I\text{-}ORG, E\text{-}ORG, O, S\text{-}LOC, B\text{-}PER, \dots\}$, where *B-*, *I-*, *E-*, *S-*, and *O* tags respectively indicate the beginning, intermediate, ending position of the entities, entity with a single word, and other types. *ORG, LOC, and PER* are categorical abbreviations of organization, location, and person, respectively. Therefore, we formulate it modeled as a sequence labeling problem, that is, $f : w_1 w_2 \cdots w_n \rightarrow y_1 y_2 \cdots y_n$. Figure 2 gives an overview of our proposed work, which is depicted in detail in the following subsections.

3.1 Embedding Layer

For a given input word sequence $\mathbf{w}$, we represent each token in the sentence by adopting both word embedding and character embedding [Lample *et al.*, 2016]. From a word sequence, we obtain the word embedding of the i-th word as follows:

$$\mathbf{x}_i^w = e^w(w_i) \tag{1}$$

where e^w denotes a pretrained word embedding lookup table. In addition, the embedding of each character within the i-th word is denoted as follows:

$$\mathbf{x}_{ij}^c = e^c(c_j) \tag{2}$$

where e^c denotes the character embedding lookup which is randomly initialized.

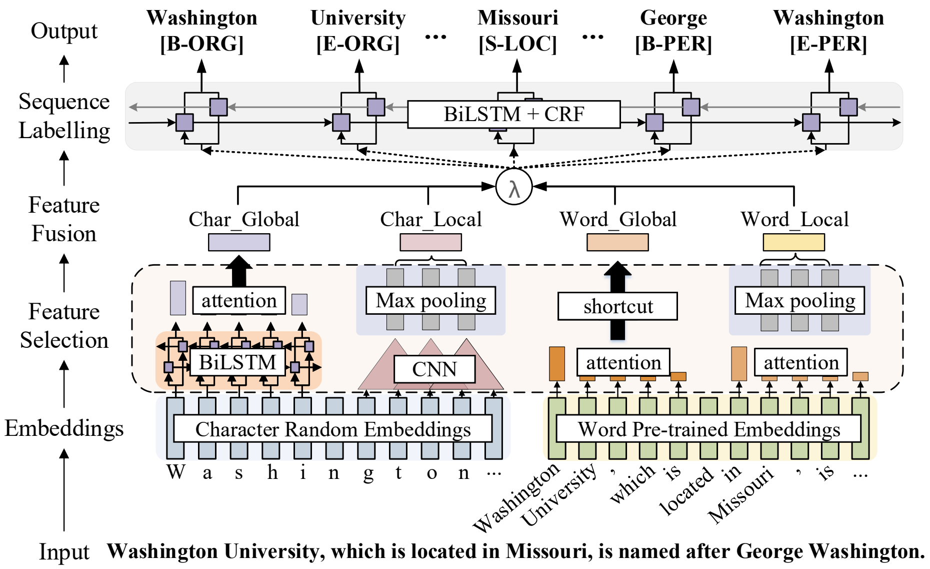

Figure 2: The attention-based multi-level feature fusion framework. Character embeddings and word embeddings of the sentence are taken as input for the feature selection layer. From this layer, we simultaneously distill character-level local feature (Char_Local), character-level global feature (Char_Global), word-level local feature (Word_Local), and word-level global feature (Word_Global). For convenience, we leave out the words labelled with O. Dashed arrows indicate a dropout operation is applied.

3.2 Multi-level Feature Selection

Global Character-level Feature Selection

As demonstrated by the BiLSTM-CRF model [Huang *et al.*, 2015], long-distance dependencies are important for NER, e.g., 'Washington' is relevant to both the past and future contexts, e.g., 'University' and 'George' in Figure 1. As the attention mechanism can relieve the limitation of encoding all information equally [Bahdanau *et al.*, 2014], we combine the BiLSTM network with the attention mechanism to facilitate NER in extracting global character-level features. Taking character embeddings $\mathbf{x}_t^c$ as input at time step t, the contextual hidden state of BiLSTM can be expressed as follows:

$$\mathbf{h}_t^{char} = [\overrightarrow{\mathbf{h}}_t^{char}; \overleftarrow{\mathbf{h}}_t^{char}] \tag{3}$$

where $\overrightarrow{\mathbf{h}}_t^{char}$ and $\overleftarrow{\mathbf{h}}_t^{char}$ denote the forward and backward ouputs of BiLSTM at time step t. Besides, we adopt the self-attention mechanism to effectively capture the relations between any two representations regardless of their distance [Vaswani *et al.*, 2017], e.g., 'Washington/*B-ORG*' is relevant but different from 'Washington/*E-PER*' in Figure 1. Concretely, we take $\mathbf{h}_t^{char}$ as the input to obtain the global character-level representation $\mathbf{h}_t^{CG}$ as follows:

$$\mathbf{h}_t^{CG} = \tanh(\mathbf{W}_c[\mathbf{c}_t; \mathbf{h}_t^{char}]) \tag{4}$$

$$\mathbf{c}_t = \sum_s \alpha_{ts}\mathbf{h}_s^{char} \tag{5}$$

$$\alpha_{ts} = softmax(\boldsymbol{\mu}_a^T \tanh(\mathbf{W}_1\mathbf{h}_s^{char} + \mathbf{W}_2\mathbf{h}_t^{char})) \tag{6}$$

Where $\mathbf{c}_t$ is the context vector, and let $\mathbf{h}_s^{char} = \mathbf{h}_t^{char}$ to obtain the additive attention weight α_{ts}. $\mathbf{W}_1$, $\mathbf{W}_2$, and $\mathbf{W}_c$ are weight matrices, and $\boldsymbol{\mu}_a$ is a vector of parameters, which are randomly initialized.

Local Character-level Feature Selection

As demonstrated by BiLSTM-CNN [Ma and Hovy, 2016], convolutional neural networks (CNNs) have been proved to be useful in capturing character-level information, such as capitalization. Due to their sparse connectivity and parameter sharing, CNNs are able to effectively process the sequences in the current receptive field akin to the attention mechanism. Furthermore, the max pooling operation can be a great boost to capture the most significant feature [Kim *et al.*, 2016], which is why they are adopted together to capture local character-level features.

We employ CNN with a redundant position of input sequences masked to extract the character-level features, which can be expressed as follows:

$$Conv(\mathbf{x}_t^c) = Mask(\mathbf{x}_t^c) * \mathbf{U} \tag{7}$$

where $\mathbf{U}$ is the filter with filter width k set as 3. The convolution operation is typically denoted with an asterisk, and the masking function, *Mask*, simply sets the padded position of input sequences as zero.

Additionally, the max pooling operation, *Max*, is applied to capture the significant local features assigned with the highest value for a given filter [Kim *et al.*, 2016], such as the capitalization of 'M' for 'Missouri'. At time step t, the character-level representation from local view is obtained as follows:

$$\mathbf{h}_t^{CL} = Max(Conv(\mathbf{x}_t^c)) \tag{8}$$

Global Word-level Feature Selection

As shown in previous works [Akbik *et al.*, 2018; Yijin Liu and Zhou, 2019], word embeddings, especially pretrained word embeddings, play an important role in capturing word similarity and relations with other words. Therefore, global word-level features, such as lexical phrases where words frequently co-occur, can be obtained by only using self-attention, which has the advantage of modeling dependencies between words without regard to their distance [Vaswani *et al.*, 2017], e.g., Label *LOC* frequently occurs after 'in'. Thus, we simply use basic dot-product attention as follows:

$$Att(\mathbf{Q},\mathbf{K},\mathbf{V}) = softmax(\mathbf{Q}\mathbf{K}^T)\mathbf{V} \tag{9}$$

where query vectors $\mathbf{Q} \in \mathbb{R}^{n\times d_w}$, key vectors $\mathbf{K} \in \mathbb{R}^{n\times d_w}$ and value vectors $\mathbf{V} \in \mathbb{R}^{n\times d_w}$. d_w denotes the dimension of each word embedding. It is noted that attention was computed without scaling to keep the scale in line with other representations. By setting $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{x}_t^w$ at time step t, the word representation based on self-attention is obtained as follows:

$$\mathbf{h}_t^{WG} = Att(\mathbf{x}_t^w, \mathbf{x}_t^w, \mathbf{x}_t^w) \tag{10}$$

In addition, we also tried to incorporate the BiLSTM network, however, this makes the result worse (e.g., AMFF* in Table 2). Therefore, we simply take $\mathbf{h}_t^{WG}$ as a shortcut for improving the gradient's backpropagation inspired by residual networks [He *et al.*, 2016].

Local Word-level Feature Selection

Inspired by the language model [Kim *et al.*, 2016], the max pooling operation facilitates the selection of prominent features, for example, we can distill local word-level features from inter-word relations based on the attention mechanism, such as the relevant keyword 'in' from the input sequence in Figure 1. Based on Equation (10), the final representation of local word embeddings $\mathbf{h}_t^{WL}$ is obtained as follows:

$$\mathbf{h}_t^{WL} = Max(FFN(\mathbf{h}_t^{WG})) \tag{11}$$

where *Max* indicates the max pooling, and *FFN* is a feed-forward network.

3.3 Multi-level Feature Fusion

Multi-level feature fusion for the NER task is a robust and efficient strategy, which can take advantage of the most significant features to achieve better results. Feature fusion for NER aims to combine multiple relevant features into a global informative representation of original input sequences. During the fusing phase, we employ a concatenation strategy to fuse the multi-level features with an automatic adjustment. For conciseness, the final fusion representation of the multi-level features is obtained as follows:

$$\mathbf{Z} = \lambda_1\mathbf{h}_t^{CG} + \lambda_2\mathbf{h}_t^{CL} + \lambda_3\mathbf{h}_t^{WG} + \lambda_4\mathbf{h}_t^{WL} \tag{12}$$

where $\mathbf{h}_t^{CG}$, $\mathbf{h}_t^{CL}$, $\mathbf{h}_t^{WG}$, and $\mathbf{h}_t^{WL}$ represent the features extracted from the above components. λ_i ($i \in \{1,2,3,4\}$) controls the degree of the importance for each component, which is randomly initialized. Moreover, this equation can be easily extended to other cases by adding more features.

3.4 Sequence Labeling for Final Prediction

The fusion representation with multi-level features is fed into a BiLSTM network to make full use of all the semantic and syntactic information at a higher level. In addition, CRF boosts the performance of NER by taking the neighbor labels into consideration to avoid mislabeling. For example, *I-ORG* can not follow *E-ORG* in the NER task with BIOES annotation. Therefore, we incorporate a CRF in the BiLSTM network to jointly decode the best chain of labels.

Formally, we suppose the fusion representation output from BiLSTM is $\mathbf{r} = (\mathbf{r_1}, \mathbf{r_2}, \ldots, \mathbf{r_n})$, with the corresponding generic label sequence $\hat{\mathbf{y}} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)$. Given the input sequence $\mathbf{r}$, the conditional probability [Ma and Hovy, 2016] is defined as $p(\mathbf{y}|\mathbf{r};\mathbf{W},\mathbf{b})$ in CRF models as follows:

$$p(\hat{\mathbf{y}}|\mathbf{r};\mathbf{W},\mathbf{b}) = \frac{\prod_{i=1}^{n}\psi_i(\hat{y}_{i-1},\hat{y}_i,\mathbf{r})}{\sum_{y'\in\mathcal{S}(\mathbf{r})}\prod_{i=1}^{n}\psi_i(y'_{i-1},y'_i,\mathbf{r})} \tag{13}$$

where y' represents an arbitrary label of all possible ones ($\mathcal{S}(\mathbf{r})$), and $\psi_i(y_{i-1}, y_i, \mathbf{r}) = exp(W_{y_{i-1},y_i}\mathbf{r}_i + b_{y_{i-1},y_i})$ where W_{y_{i-1},y_i} and b_{y_{i-1},y_i} are the weight parameter and bias parameter corresponding to label pair (y_{i-1}, y_i).

For CRF training, the objective of the model is to maximize the following log-likelihood, which is given by:

$$L(\mathbf{W},\mathbf{b}) = \sum_i \log(p(\hat{\mathbf{y}}|\mathbf{r};\mathbf{W},\mathbf{b})) \tag{14}$$

During the decoding phase, we search for the best label sequence y^* that maximizes the likelihood as follows:

$$y^* = \underset{\hat{y}\in\mathcal{S}(\mathbf{r})}{\arg\max}\, p(\hat{\mathbf{y}}|\mathbf{r};\mathbf{W},\mathbf{b}) \tag{15}$$

Furthermore, for sequence labeling, we adopt Viterbi to calculate the final tag sequence efficiently.

4 Experiments

4.1 Datasets and Baseline Methods

To verify the effectiveness of the proposed framework, we conduct experiments on the following four datasets, CoNLL-2003 [Sang and De Meulder, 2003], NCBI-disease [Doğan *et al.*, 2014], SciERC [Luan *et al.*, 2018], and JNLPBA [Kim *et al.*, 2004]. All datasets have been separated into train/develop/test sets, including 4/1/6/5 entity types, respectively. Table 1 presents some statistics of the 4 datasets.

We compare our proposed model with the following classic (i.e., BiLSTM-CRF, BiLSTM-CNNs, and NeuralNER) and state-of-the-art (i.e., CS Embeddings, SciBERT, and CollaboNet) methods, respectively:

- BiLSTM-CRF [Huang *et al.*, 2015]: This applies the BiLSTM network to learn both past and future features of word embeddings with CRF for sequence labeling.

Dataset		train	develop	test
CoNLL-2003	#tok	204567	51578	46666
	#ent	23499	5942	5648
NCBI-disease	#tok	135701	23969	24497
	#ent	5134	787	960
SciERC	#tok	45762	6571	13501
	#ent	5572	808	1683
JNLPBA	#tok	441905	50646	101039
	#ent	46390	4911	8662

Table 1: Statistics of these four datasets, #tok denotes tokens and #ent denotes entities.

Model	CoNLL-2003			NCBI-disease			SciERC			JNLPBA		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
BiLSTM-CRF (2015)	92.78	87.43	90.02	85.47	74.32	79.51	67.83	47.83	56.09	73.47	68.27	70.77
BiLSTM-CNNs (2016)	91.35	91.06	91.21	82.61	76.67	79.52	68.01	50.18	57.75	73.96	70.52	72.20
NeuralNER (2016)	90.88	90.62	90.75	85.67	64.30	73.46	67.43	47.15	55.49	73.08	71.56	72.31
CS Embeddings (2018)	92.37	93.12	92.74	85.02	87.33	86.16	62.58	61.99	62.28	71.18	77.68	74.29
SciBERT (2019)	88.46	89.13	88.79	84.32	89.06	86.63	63.83	65.42	64.61	70.73	80.36	75.24
CollaboNet (2019)	87.31	81.47	84.29	80.50	81.42	80.95	64.32	56.43	60.12	72.92	82.42	77.38
AMFF*	94.95	90.74	92.80	90.23	85.62	87.86	67.90	57.33	62.17	80.37	79.69	80.03
AMFF(without Attention)	94.20	93.06	93.63	87.90	89.03	88.46	67.87	61.03	64.27	78.60	79.79	79.72
AMFF	94.83	94.12	**94.48**	89.60	94.76	**92.11**	71.01	65.86	**68.34**	79.09	81.99	**80.51**

Table 2: Experiment results on four benchmark datasets compared to the classic and state-of-the-art methods. Standard precision (P), recall (R), and F1 score (F1) are employed as evaluation metrics. AMFF* indicates the BiLSTM network has been incorporated into the global word-level component.

- BiLSTM-CNNs [Chiu and Nichols, 2016]: This extracts character-level features using CNN, and word-level features from pretrained word embeddings, in addition to encoding partial lexicon matches in neural networks.
- NeuralNER [Lample *et al.*, 2016]: Similar to Chiu [Chiu and Nichols, 2016], this regards words as sequences of character and learn character-level features from a BiLSTM, rather than CNNs.
- CS Embeddings [Akbik *et al.*, 2018]: It obtains context embeddings at character level, and the final representation is concatenated with pretrianed word embeddings.
- SciBERT (Beltagy2019) [Beltagy *et al.*, 2019]: This introduces a contextualized embedding model for scientific text based on BERT, which achieves the state-of-the-art on several tasks.
- CollaboNet (Yoon2019) [Yoon *et al.*, 2019]: This is built upon multiple identical single-task NER models (STMs) that send information to the proper model for more accurate predictions in the biomedical field.

In the experiment, we take both the pretrained word embedding GloVe [Pennington *et al.*, 2014] and the randomly initialized character embedding as input, and train our proposed model using SGD to perform backpropagation through time. To avoid overfitting, dropout is applied to the input of each component as well as the output of the feature fusion layer in our model. For time efficiency, we merely initialize the related hyper-parameter values according to the aforementioned baselines. We repeat the experiment with an early stopping strategy 10 times, and report the average performance on the test set as the final performance. We adopt the F1 metric and BIOES tagging scheme for all datasets.

4.2 Overall Results and Comparisons

Table 2 gives the experiment results of AMFF and the baseline methods. For a fair comparison, we report their average results on the four benchmark datasets. Classic character-based and word-based NER methods obtain lower F1 scores than recent methods on most benchmark datasets, because the final prediction in the BiLSTM-CRF network may be misled by the previous label due to insufficient information. To the best of our knowledge, SciBERT achieves state-of-the-art performance on NCBI-disease and SciERC, which depends on pretraining on a large corpus of scientific publications to generate contextualized embeddings. CollaboNet obtains the best result on JNLPBA due to multi-task learning, which may make a wrong prediction when an error is overlapped. However, classic methods achieve an F1 score of more than 90 on CoNLL-2003, which is slightly better than the result of SciBERT and CollaboNet. This is probably because these two methods are designed for academic and biomedical fields, respectively, which fail to capture general features such as local character features and lexical phrases from given sequences. In addition, CS Embeddings achieves the state-of-the-art with an F1 score of 92.74 on CoNLL-2003 and obtains competitive results on the other datasets, which merely takes the global word- and character-level features into consideration which is similar to a part of our proposed AMFF.

As shown in Table 2, our proposed method achieves the best results on the four benchmark datasets , which verifies its effectiveness. Specifically, obtaining global word-level features merely based on attention contributes to overall performance (compared with AMFF*), which might because pretrained word embeddings have already provided inter-word relations in no need of BiLSTM. In addition, there would be a slight performance degradation if we get rid of attention from our model, which highlights the importance of incorporating multi-level features based on attention mechanisms.

4.3 Ablation Study

As Table 3 shows, the AMFF with a single component is not good enough while the one with multiple components fusion tends to be more competitive, which might because the multi-level features captured from the four primary components (i.e., CG, CL, WG, and WL) contribute to the effectiveness of our model. Based on the attention mechanism, word-level components seem to be more effective than character-level components, which should be credited to the pretrained word embedding. For SciERC, when fusing three components, the F1 score decreases to 65.86, which could be due to the noise caused by the fusion components. However, in general, our proposed model tends to be more effective and robust with the number of fusion components increasing. This is mainly because all these components help to improve performance from multi-level perspectives.

Component	CoNLL-2003			NCBI-disease			SciERC			JNLPBA		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
AMFF_CG	49.14	30.19	37.40	73.84	40.01	51.90	36.02	18.61	24.54	56.08	56.55	56.31
AMFF_CL	73.17	70.31	71.71	85.15	84.60	84.87	53.94	28.17	37.01	76.56	69.73	72.99
AMFF_WG	92.94	91.25	92.09	85.47	81.19	83.28	64.86	57.12	60.74	75.91	76.22	76.07
AMFF_WL	93.20	91.97	92.58	87.22	86.20	86.71	64.91	64.01	64.46	78.33	71.45	74.74
AMFF_WG_CG	93.43	92.28	92.85	87.03	86.89	86.96	68.47	64.52	66.44	77.15	79.43	78.27
AMFF_WG_CG_CL	93.40	92.54	92.96	88.96	85.40	87.14	68.13	63.75	65.87	77.49	80.95	79.18
AMFF_WG_CG_WL_CL	94.57	94.38	**94.48**	91.22	95.01	**93.08**	75.53	65.55	**70.19**	82.88	81.02	**81.94**

Table 3: Results of the ablation study of the AMFF framework on the develop set. 'WG', 'WL', 'CG', and 'CL' denote the global word-level component, local word-level component, global character-level component, and local character-level component, respectively.

Sentence	Washington University, which is located in Missouri, is named after George Washington.
Gold Label	Washington University: [ORG]; Missouri: [LOC]; George Washington: [PER]
CS Embeddings	Washington: [B-ORG], [B-PER]; University: [E-ORG], [E-PER]; Missouri: [S-LOC]; George: [B-PER]; Washington: [E-PER]
SciBERT	Washington: [B-ORG], [B-PER]; University: [E-ORG], [E-PER]; Missouri: [S-LOC]; George: [B-PER]; Washington: [E-PER]
AMFF	**Washington**: [B-ORG]; University: [E-ORG] ; Missouri: [S-LOC]; George: [B-PER]; **Washington**: [E-PER]

Table 4: Case study. The bold words attract more attention.

4.4 Case Study for Detailed Analysis

Table 4 shows a case study comparing our model with CS Embeddings [Akbik *et al.*, 2018] and SciBERT[Beltagy *et al.*, 2019] , which are more representative than the others. In the example, the polysemous word 'Washington' is likely to lead to ambiguity, i.e., the first 'Washington' denotes an organization (*ORG*) together with 'University' while the second should be categorized as a person (*PER*) in 'George Washington'. CS Embeddings and SciBERT may recognize 'Washington' as *B-ORG* or *B-PER* from the context, which causes 'University' to be erroneously labeled *E-PER* due to a lack of other auxiliary features. Different from the existing methods, the AMFF can easily recognize entities by taking more features of the original sequences into consideration, such as the lexical phrase and the keyword 'in', which is vital for distinguishing entity categorical labels. Furthermore, the AMFF put more emphasis on disambiguation based on self-attention, which enables the long-distance dependence to be captured, as shown in Figure 1. This demonstrates that our model has the advantage of distinguishing polysemous words by fusing multi-level features from different perspectives.

4.5 Parameter Sensitivity Analysis

Four primary parameters, i.e., dropout rate, LSTM size, filter number, and batch size, are selected to verify the effect of parameters on the effectiveness of the AMFF. The dropout rate denotes what percentage of units will be dropped in a neural network, the LSTM size controls how many hidden state units there are in sequence labeling, the number of filters affects the output shape of the character-level CNN module, and the batch size controls training efficiency and the allocated resources. To study uncertainty in the output of our proposed model, we employ single-parameter sensitivity analysis by varying one parameter while fixing the others each time. As shown in Figure 3, the AMFF keeps high performance while the parameter varies on the four benchmark datasets, which demonstrates that multi-level features contribute to NER, and further verifies the effectiveness and robustness of our proposed model.

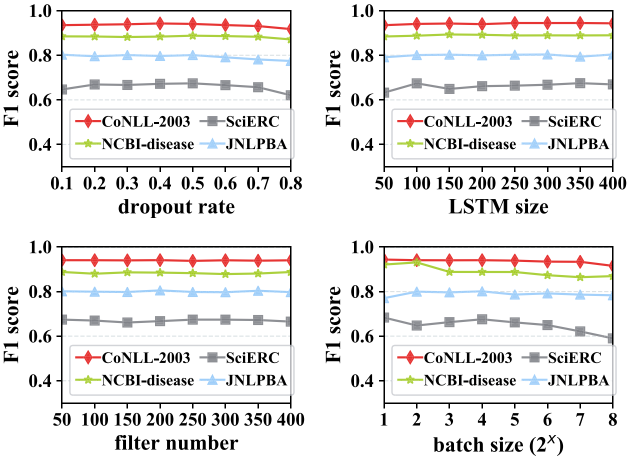

Figure 3: Parameter sensitivity analysis of the AMFF framework.

5 Conclusion

This paper presents a novel AMFF framework, which effectively leverages multi-level features to predict entity categorical labels. The proposed framework captures character-level and word-level features from both global and local perspectives, e.g., capitalization, inter-word relations, keywords, and lexical phrases, by adopting attention mechanisms from different perspectives. Furthermore, the proposed framework can be easily extended by incorporating more features, such as affixe, to boost the performance of NER. The Experiment results demonstrate that the AMFF surpasses previous state-of-the-art models on CoNLL-2003, NCBI-disease, SciERC, and JNLPBA datasets.

Acknowledgments

This work is partially supported by National Natural Science Foundation of China through grants No.61976102, No.U19A2065, and No.61902145. This work is also partially supported by NSF through grant IIS-1763365 and by FSU.

References

[Akbik *et al.*, 2018] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *COLING*, pages 1638–1649, 2018.

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint:1409.0473*, 2014.

[Beltagy *et al.*, 2019] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. In *EMNLP*, pages 3606–3611, 2019.

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

[Collobert *et al.*, 2011] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(1):2493–2537, 2011.

[Doğan *et al.*, 2014] Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014.

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

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

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

[Kim *et al.*, 2004] Jin-Dong Kim, Tomoko Ohta, Yoshimasa Tsuruoka, Yuka Tateisi, and Nigel Collier. Introduction to the bio-entity recognition task at jnlpba. In *JNLPBA*, pages 70–75. Citeseer, 2004.

[Kim *et al.*, 2016] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. In *AAAI*, pages 2741–2749, 2016.

[Kuru *et al.*, 2016] Onur Kuru, Ozan Arkan Can, and Deniz Yuret. Charner: Character-level named entity recognition. In *COLING*, pages 911–921, 2016.

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

[Liu *et al.*, 2018] Liyuan Liu, Jingbo Shang, Xiang Ren, Frank Fangzheng Xu, Huan Gui, Jian Peng, and Jiawei Han. Empower sequence labeling with task-aware neural language model. In *AAAI*, pages 5253–5260, 2018.

[Luan *et al.*, 2018] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. *arXiv preprint:1808.09602*, 2018.

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

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

[Quimbaya *et al.*, 2016] Alexandra Pomares Quimbaya, Alejandro Sierra Múnera, Rafael Andrés González Rivera, Julián Camilo Daza Rodríguez, Oscar Mauricio Muñoz Velandia, Angel Alberto Garcia Peña, and Cyril Labbé. Named entity recognition over electronic health records through a combined dictionary-based approach. *Procedia Computer Science*, 100:55–61, 2016.

[Rahem and Omar, 2014] Khmael Rakm Rahem and Nazlia Omar. Drug-related crime information extraction and analysis. In *ICIMU*, pages 250–254. IEEE, 2014.

[Sang and De Meulder, 2003] Erik F Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*, 2003.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[Xin *et al.*, 2018] Yingwei Xin, Ethan Hart, Vibhuti Mahajan, and Jean-David Ruvini. Learning better internal structure of words for sequence labeling. In *EMNLP*, pages 2584–2593, 2018.

[Yadav *et al.*, 2018] Vikas Yadav, Rebecca Sharp, and Steven Bethard. Deep affix features improve neural named entity recognizers. In **SEM*, pages 167–172, 2018.

[Yijin Liu and Zhou, 2019] Jinchao Zhang Jinan Xu Yufeng Chen Yijin Liu, Fandong Meng and Jie Zhou. Gcdt: A global context enhanced deep transition architecture for sequence labeling. In *ACL*, pages 2431–2441, 2019.

[Yoon *et al.*, 2019] Wonjin Yoon, Chan Ho So, Jinhyuk Lee, and Jaewoo Kang. Collabonet: collaboration of deep neural networks for biomedical named entity recognition. *BMC bioinformatics*, 20(10):249, 2019.

[Zheng *et al.*, 2019] Changmeng Zheng, Yi Cai, Jingyun Xu, Ho-fung Leung, and Guandong Xu. A boundary-aware neural model for nested named entity recognition. In *EMNLP*, pages 357–366, 2019.