Context-Aware Attentive Multilevel Feature Fusion for Named Entity Recognition

Zhiwei Yang, Jing Ma, Hechang Chen, Jiawei Zhang, and Yi Chang, *Senior Member, IEEE*

***Abstract*— In the era of information explosion, named entity recognition (NER) has attracted widespread attention in the field of natural language processing, as it is fundamental to information extraction. Recently, methods of NER based on representation learning, e.g., character embedding and word embedding, have demonstrated 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, and lexical phrases, from multilevel perspectives. Intuitively, multilevel features can be helpful when recognizing named entities from complex sentences. In this study, we propose a novel attentive multilevel feature fusion (AMFF) model for NER, which captures the multilevel features in the current context from various perspectives. It consists of four components to, respectively, capture the local character-level (CL), global character-level (CG), local word-level (WL), and global word-level (WG) features in the current context. In addition, we further define document-level features crafted from other sentences to enhance the representation learning of the current context. To this end, we introduce a novel context-aware attentive multilevel feature fusion (CAMFF) model based on AMFF, to fully leverage document-level features from all the previous inputs. The obtained multilevel features are then fused and fed into a bidirectional long short-term memory (BiLSTM)-conditional random field (CRF) network for the final sequence labeling. Extensive experiments on four benchmark datasets demonstrate that our proposed AMFF and CAMFF models outperform a set of state-of-the-art baseline methods and the features learned from multiple levels are complementary.**

Manuscript received May 8, 2021; revised February 13, 2022 and March 14, 2022; accepted May 19, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61976102, Grant U19A2065, and Grant 61902145; in part by the National Science Foundation (NSF) under Grant IIS-1763365 and Grant IIS-2106972; and in part by the University of California at Davis and Hong Kong Baptist University (HKBU) One-Off Tier 2 Start-Up Grant RCOFSGT2/20-21/SCI/004. *(Corresponding authors: Hechang Chen; Yi Chang.)*

Zhiwei Yang is with the College of Computer Science and Technology and the Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China (e-mail: yangzw18@mails.jlu.edu.cn).

Jing Ma is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, China (e-mail: majing@comp.hkbu.edu.hk).

Hechang Chen is with the School of Artificial Intelligence and the Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China (e-mail: chenhc@jlu.edu.cn).

Jiawei Zhang is with the IFM Laboratory, Department of Computer Science, University of California at Davis, Davis, CA 95616 USA (e-mail: jiawei@ifmlab.org).

Yi Chang is with the School of Artificial Intelligence, the International Center of Future Science, and the Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China (e-mail: yichang@jlu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3178522.

Digital Object Identifier 10.1109/TNNLS.2022.3178522

***Index Terms*— Attention mechanism, multilevel feature extraction, named entity recognition (NER), sequence labeling.**

I. Introduction

NAMED entity recognition (NER) is a fundamental task of information extraction to identify entities from raw text and assign them predefined tags, such as person (PER), organization (ORG), and location (LOC) [1]. NER has been extensively studied for various tasks such as part-of-speech tagging, chunking, and semantic role labeling [2]. Considering the diversity and complexity of natural language, named entities can be characterized by multiple features (character-level and word-level) and from multiple perspectives (local and global), as illustrated in Fig. 1. The example in the figure illustrates that to accurately label polysemous words (e.g., Washington), NER should consider capitalization ("W"), keywords ("in"), and lexical phrases.

Previous knowledge-based approaches for NER merely depended on handcrafted rules and domain-specific dictionaries to recognize named entities [3], [4]. However, such approaches are manual and thus prone to poor coverage. Similarly, traditional machine learning approaches used supervised learning by incorporating a wide variety of hand-crafted features. To alleviate the heavy manual effort associated with these approaches, neural models were proposed to learn the implicit features by utilizing word-level embedding [5], character-level embedding [6], or both [7]. However, these methods largely ignore or oversimplify the correlations among the different levels of features such as word-level and character-level features. Although NER approaches combining word-level embedding and character-level embedding have demonstrated improved results [7]–[9], they may pay little attention to fusing multifeatures and therefore lose a significant amount of information. For example, when only take features of adjacent words or characters in the current context into consideration without incorporating character-level global features, the polysemous word "Washington" in "Washington University" (ORG) might be mislabeled as *B*-PER in the sentence in Fig. 1. This may result in "Washington University" being misclassified as PER.

To the best of our knowledge, no existing NER model incorporates such multilevel features except that outlined in our recent work [10]. That work was a preliminary study on fusing multilevel semantic and syntactic features for identifying named entities in an input sentence. The proposed model included four parallel attention-based components and integrated multilevel features from different perspectives based on

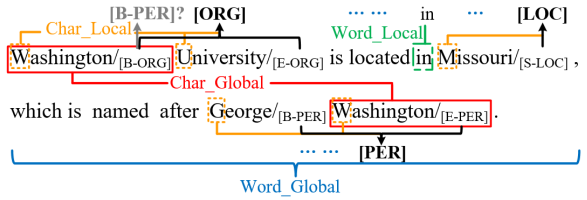

Fig. 1. Brief illustration of the multilevel features. There are four kinds of character-level and word-level features from local and global perspectives: Char_Local, Char_Global, Word_Local, and Word_Global, such as the capitalization (in orange), the polysemous word "Washington" (in red), the keyword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

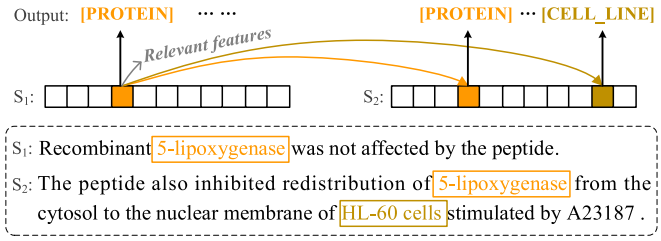

Fig. 2. Illustration of extended context in the biomedical literature. Previous context can be incorporated into the current context to enhance NER, e.g., the feature relevant to the entity "5-lipoxygenase" in the sentence S_1 may help the recognition of entities that are identical or with similar features in the sentence S_2.

the current context, thereby achieving promising results. The current study extends the previous work in [10] by: 1) introducing a memory-distilled network to memorize and distill previous contextual features for enhancing representation learning enhancement and 2) performing more in-depth experiments and analyses based on the CoNLL-2003, NCBI-disease, SciERC, and JNLPBA datasets. The results demonstrate that the new NER model exhibits superior performance.

In this article, we first propose the attentive multilevel feature fusion (AMFF) framework for NER, where the multilevel semantic and syntactic features of a given input sequence are simultaneously captured from different views, as shown in Fig. 1. Inspired by the transformer network [11], we explicitly employ four components, namely the local character-level (Char_Local), global character-level (Char_Global), local word-level (Word_Local), and global word-level (Word_Global) components, to process the current input sequence. The fusion representation of the context-aware multilevel features is then fed into the bidirectional long short-term memory (BiLSTM)-conditional random field (CRF) network [5] for the final prediction.

However, a drawback of this model is that it only considers the current context, which prevents the utilization of previous contextual information. For example, as shown in Fig. 2, the feature relevant to the entity "5-lipoxygenase" extracted from the previous context S_1 could enhance the recognition of the identical entity "5-lipoxygenase" (PROTEIN) or other entities [e.g., "HL-60 cells" (CELL_LINE)] in S_2 with similar features such as those in the combination of characters and numbers denoting proteins or cells in biology. Therefore, a more general framework for NER is urgently needed.

Previous studies have found that NER can be enhanced by incorporating document-level information [12]. In this work, we further propose the context-aware AMFF (CAMFF) model, which memorizes and distills context-aware document-level features, i.e., character-level and word-level features captured from other sentences in the document. To distill context-aware document-level features, we design a set of dilated convolutional neural networks with different dilated rates from fine to coarse. Besides, previous contextual features are concatenated to the current context iteratively, thereby broadening the contextual information in sequence labeling.

We conduct extensive experiments on four datasets, i.e., CoNLL-2003, SciERC, NCBI-disease, and JNLPBA, and demonstrated that: 1) the proposed AMFF yields outstanding improvements over the state-of-the-art baseline methods; 2) the CAMFF framework is more effective at capturing multilevel features; and 3) the document-level features are complementary with the local/global character/word-level features. The main contributions of our article are fourfold.

1) We propose the AMFF framework for NER, which enables the multilevel features from diverse word-level and character-level perspectives to be integrated. Based on AMFF, we develop CAMFF by incorporating previous contextual features to broaden the scope of the context. To the best of our knowledge, this is the first study to use attention mechanisms for capturing multilevel contextual features from different perspectives.
2) By adopting feature selectors for local character-level (CL), global character-level (CG), local word-level (WL), and global word-level (WG) in AMFF to capture the features pertaining to capitalization, inter-word relations, keywords, and lexical phrases, respectively. We simplify the problem and improve interpretability.
3) In CAMFF, we extend the scope of the current context by incorporating previous contextual features iteratively. This provides a novel solution for aggregating effective information from an extended context for NER.
4) Extensive validations on four benchmark datasets against the state-of-the-art models demonstrate the superiority of our proposed methods. Systematic analyses reveal an in-depth understanding of each component and the robustness of the proposed frameworks. Moreover, additional experiments show that the previous context contributes to the effectiveness and robustness for NER.

The rest of this article is organized as follows. Related work is first summarized in Section II. We then formulate the problem of NER in Section III. Next, the proposed AMFF model and its extension CAMFF are detailed in Section IV, as illustrated in Figs. 3 and 4, respectively. After that, Section V presents the experimental results and analyses. Finally, Section VI summarizes the conclusions and outlines directions for future work.

It is noted that there is a shorter conference version of this article published in [10]. To the best of our knowledge, most NER studies have extracted entities solely from current context fragments, thereby limiting representation learning. Our initial conference paper focused on the multilevel features in the current context for NER and was therefore limited in the scope of the contextual information it included. In this

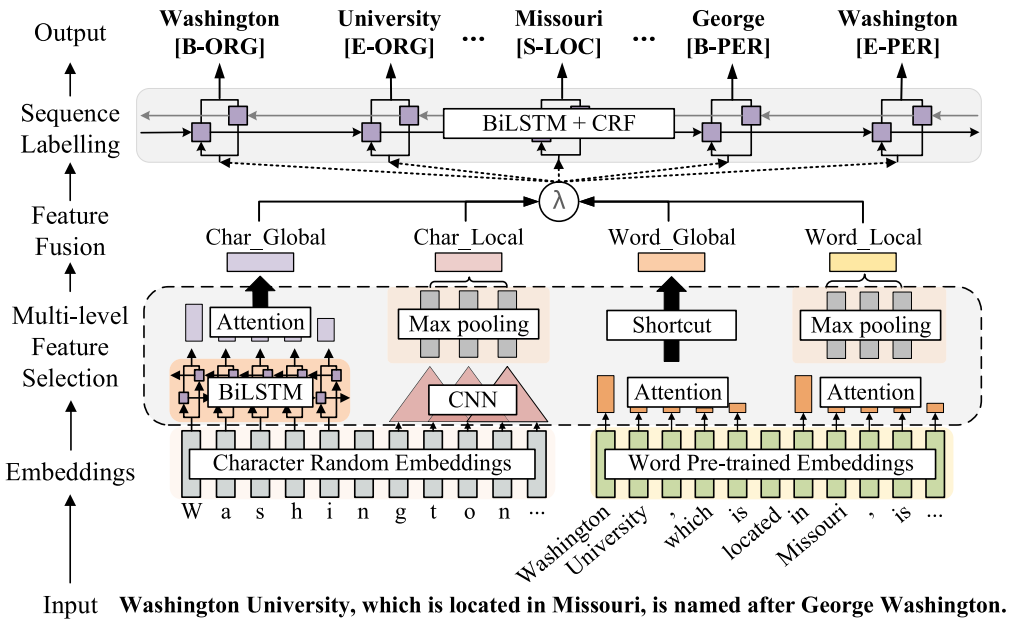

Fig. 3. AMFF framework. Character embeddings and word embeddings of the sentence are inputs for the feature selection layer. From this layer, Char_Global, Char_Local, Word_Global, and Word_Local components are simultaneously adopted to extract the character-level global (CG), character-level local (CL), word-level global (WG), and word-level local (WL) features, respectively. For convenience, we leave out the words labeled with *O*. Dashed arrows indicate that a dropout operation is applied.

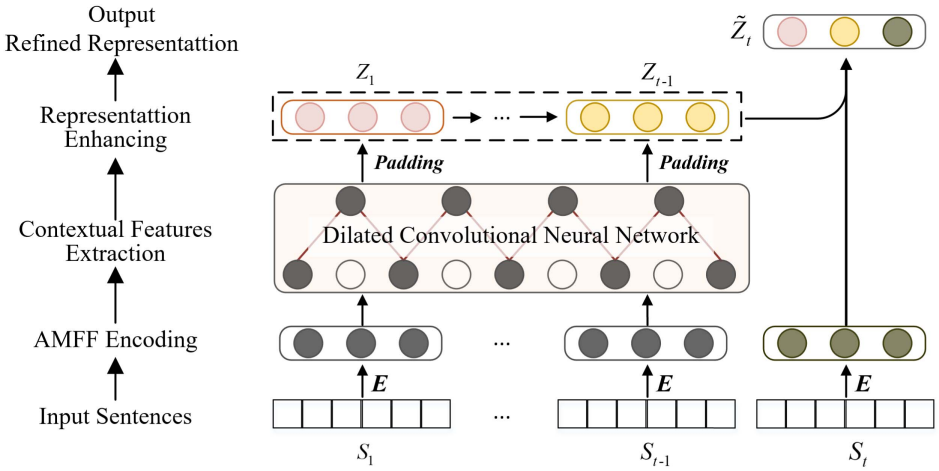

Fig. 4. Additional Contextual feature enhancing in CAMFF. *E* denotes the fused multilevel feature encoding before the sequence labeling layer of AMFF.

manuscript, we propose a remedy solution CAMFF to address this limitation and provide additional analyses on the results of our experiments.

II. Related Work

With the recent advancements in deep learning (DL), DL-based NER models are being predominantly adopted for NER and have achieved the state-of-the-art results [1], [13]. Compared with traditional approaches, DL is superior at discovering hidden features from context automatically. Existing DL-based NER methods are primarily based on word embeddings, character embeddings, and combinations of other embeddings as follows.

The first word embedding-based NER approach [2] adopted convolution neural networks (CNNs) to produce local features, and a CRF layer to predict entity attributes. To incorporate long-distance dependencies, a subsequent study replaced the CNN layer with a bi-directional long short-term memory (BiLSTM) layer, thereby enabling a better selection of global features [5]. Another study combined CNN and BiLSTM to improve the performance of linguistic sequence labeling [14]. However, these methods did not account for effects at the character level, thereby potentially losing important information.

Word representations can be obtained from character-level embeddings as a sentence can be regarded as a character sequence. A character-level convolutional neural network (CharCNN) [15] was proposed to extract local sub-word information. This approach relied on LSTM for contextual features and the softmax function for the final prediction, which highlighted the character embeddings for NER. Another study adopted character-level recurrent neural networks (CharRNN) [6] to select global features from the context, and improved NER using CRF. In addition, it achieved better performance in dealing with multiple languages [16], [17]. However, methods based on character embeddings do not

prioritize word-level features, and therefore exclude valuable information.

Recent studies have demonstrated the advantage of recognizing named entities based on a combination of embeddings [7], [18]. To capture both global and local features, existing methods have incorporated additional types of embeddings. For example, BERT-based methods such as SciBERT [9] have incorporated token embeddings, segment embeddings, and position embeddings for NER. Moreover, other auxiliary information such as affix embeddings can also be used to augment the NER architecture [19]. Another study utilized the multitask learning strategy that divides the original task into multiple subtasks for performing nested NER [20]. Although these methods have enhanced NER, few of them have explored the attention mechanism for multilevel feature selection in NER and thereby have not comprehensively utilized the information that can be leveraged.

In addition, there is another significant challenge in utilizing document-level features from previous context. As we known, LSTM language models use only 200 context words on average [21]. One study addressed this using attentive language models to improve the effect of long-term dependency on separated fixed-length segments [12]. Recent studies have adopted the attention mechanism [10], [22], [23], dependency-based BiLSTM networks [24], [25], and the purely parsing-based approach [26] to learn global dependencies in a sentence context, and these have achieved highly competitive performances. Another study [27] proposed a FOFE-based method, which encodes the left and right context of a sentence into the target span similar to bag-of-words. However, there is a paucity of work that incorporates document-level semantic features for NER. Owing to the fixed lengths of the contexts, existing models do not effectively capture features beyond the current context, thereby preventing the utilization of previous contextual information [28].

In contrast to existing models that mine information from merely one perspective, our model focuses on leveraging multilevel features from multiple perspectives. This enables obtaining more types of information and deliver a comprehensive final prediction. Our model also fully utilizes information from previous inputs by distilling previous contextual features to extend the current context, which further contributes to aggregating information from a larger scope for NER.

III. Problem Formulation

Given an input sentence S composed of a sequence of words $\{w_1, w_2, \ldots, w_t, \ldots, w_n\}$, where n is the total number of words in the sentence, we assign each word w_t with a label y_t that takes one possible class from the named entity label set: $\mathbf{y} = \{B-\text{ORG},\ I-\text{ORG},\ E-\text{ORG},\ O,$ S-LOC,B-PER,…$\}$, where the tags B-, I-, and E-, respectively, indicate the beginning, intermediate, and ending positions of the entities, the tag S- indicates an entity with a single word, and the tag O indicates other types. ORG, LOC, and PER are categorical abbreviations of organization, location, and person, respectively. Thus, we formulate it as a sequence labeling problem, that is, $f : \{w_1, w_2, \ldots, w_t, \ldots, w_n\} \rightarrow \{y_1, y_2, \ldots, y_t, \ldots, y_n\}$. Figs. 3 and 4 present the overviews of our proposed frameworks, which are depicted in Section IV.

IV. Our Proposed Framework

This section will introduce our proposed models, i.e., AMFF and CAMFF for NER, respectively. AMFF consists of shared embedding layer, multilevel feature selection layer, feature fusion layer, and sequence labeling layer, as illustrated in Fig. 3. In addition, CAMFF contains a contextual feature enhancing layer after the multilevel feature fusion layer, as illustrated in Fig. 4.

A. 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 [29]. From a word sequence, we obtain the word embedding of the tth word as follows:

$$\mathbf{x}_t^w = e^w(w_t) \tag{1}$$

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

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

where e^c denotes the character embedding lookup, which is randomly initialized in this work.

B. Multilevel Feature Selection

The multilevel feature selection contains four components, i.e., Char_Global, Char_Local, Word_Global, and Word_Local, to extract the character-level global (CG), character-level local (CL), word-level global (WG), and word-level local (WL) features, respectively, as illustrate in Fig. 3.

1) CG Feature Selection: As demonstrated by the BiLSTM-CRF model [5], long-distance dependencies are important for NER. For example, in the sentence in Fig. 1, "Washington" is relevant to both the past and future contexts, i.e., "University" and "George." As the attention mechanism eliminates the necessity for encoding all information equally [30], we combine the BiLSTM network with the attention mechanism to facilitate NER for extracting CG features. We take character embeddings in the tth word $\mathbf{x}_{tj}^c$ into BiLSTM to learn hidden states and the contextual hidden state is expressed as follows:

$$\mathbf{h}_t^{\text{char}} = \left[\overrightarrow{\mathbf{h}}_t^{\text{char}} \oplus \overleftarrow{\mathbf{h}}_t^{\text{char}}\right] \tag{3}$$

where $\overrightarrow{\mathbf{h}}_t^{\text{char}}$ and $\overleftarrow{\mathbf{h}}_t^{\text{char}}$ denote the forward and backward outputs of BiLSTM at time step t. $\oplus$ denotes concatenation. We adopt the self-attention mechanism to effectively capture the relationships between any two representations regardless of the distance between them [11], e.g., "Washington/B-ORG" is relevant to but different from "Washington/E-PER" in Fig. 1. Formally, we take $\mathbf{h}_t^{\text{char}}$ as the input to obtain the CG representation $\mathbf{h}_t^{\text{CG}}$ as follows:

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

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

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

Here $\mathbf{c}_t$ is the context vector. We then let $\mathbf{h}_s^{\text{char}} = \mathbf{h}_t^{\text{char}}$ 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.

2) CL Feature Selection: As demonstrated by BiLSTM-CNN [31], convolutional neural networks (CNNs) are useful for capturing character-level information, such as capitalization. Owing 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 significantly enhances the capturing of the most significant feature [15]. This is why BiLSTM and CNN are adopted together to capture CL features.

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

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

where $\mathbf{U}$ is the filter with filter width k set as 3. The convolution operation is typically denoted by 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 [15], such as the capitalization of "M" for "Missouri." At time step t, the character-level representation from the local perspective is obtained as follows:

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

Thus, $\mathbf{h}_t^{\text{CL}}$ represents the CL representation.

3) WG Feature Selection: Previous studies [7], [32] have shown that word embeddings, especially the pretrained embeddings, play an important role in capturing word similarity and relations in other words. Therefore, WG features, such as lexical phrases where words frequently co-occur, can be obtained by merely using self-attention, which has the advantage of modeling dependencies between words regardless of the distance between them [11]. For example, the label LOC frequently occurs after "in." We simply use basic dot-product attention as follows:

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

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 maintain comparability with other representations. By setting $\mathbf{Q} = \mathbf{W}^Q \mathbf{x}_t^w, \mathbf{K} = \mathbf{W}^K \mathbf{x}_t^w, \mathbf{V} = \mathbf{W}^V \mathbf{x}_t^w$ where $\mathbf{W}$ is the parameter to be learned, the word representation based on self-attention is obtained as follows:

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

In our experiments, we found that incorporating the BiLSTM network worsened the result (e.g., AMFF-BI in Table II). Therefore, inspired by residual networks [33], we simply use $\mathbf{h}_t^{\text{WG}}$ as a shortcut connection for improving the gradient's back-propagation.

4) WL Feature Selection: Inspired by the language model [15], the max pooling operation facilitates the selection of prominent features. For example, we can distill WL features from inter-word relations based on the attention mechanism, such as the relevant keyword "in" from the input sequence in Fig. 1. Based on (10), the final representation of local word embeddings $\mathbf{h}_t^{\text{WL}}$ is obtained as follows:

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

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

C. Multilevel Feature Fusion

Multilevel feature fusion is a robust and efficient strategy for NER, as it leverages the most significant features to achieve better results. The objective of feature fusion for NER is to form representations of the original input sequences based on global information by combining multiple relevant features. We employ a concatenation strategy to fuse the multilevel features with finetuning. For conciseness, the final fusion representation of the multilevel features from the current context is obtained as follows:

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

where $\mathbf{h}_t^{\text{CG}}$, $\mathbf{h}_t^{\text{CL}}$, $\mathbf{h}_t^{\text{WG}}$, and $\mathbf{h}_t^{\text{WL}}$ represent the features extracted from the above components, respectively. λ_m ($m \in \{1, 2, 3, 4\}$) 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 relevant features.

D. Additional Contextual Feature Enhancing

To our knowledge, almost all approaches for NER extracted entities solely from a fragment of the current context, which limits representation learning. The success of [12] demonstrates that widening the scope of the utilized information enhances contextual representation learning in NER. For example, in the sentences in Fig. 2, the entities "5-lipoxygenase" and "HL-60 cells" are most likely to be projected on to the vicinity of a feature that indicates a combination of characters and numbers in the field of biology. Since AMFF only encodes multilevel features from the current context, the natural extension is to fully utilize features from previous contexts. This enhances NER in the current context by connecting it with similar contexts at the document level.

CAMFF is a memory-distilled network based on AMFF, which memorizes and distills contextual features from previous inputs, i.e., document-level features, to improve NER performance, as shown in Fig. 4. Thus, the proposed framework enhanced by previous contextual features is referred to as CAMFF. Unlike [12] that memorized the representations for each unique word in the sentences using a key-value memory network [34], we explicitly designed a set of dilated CNNs with different dilation rates ranging from fine to coarse. This contributes to the distillation of multigrained discriminative features from the previous contexts and obtains refined representations based on extended context.

Algorithm 1 CAMFF for NER

Input:

A sequence of words $\mathcal{S} = \{w_1, w_2, \ldots, w_n\}$;

The dimension of word embeddings d^w and character embedding d^c; Word-level LSTM units d^{lstm} and character-level LSTM units $d^{lstm'}$.

Output:

Named entity labels $\hat{\mathbf{y}}^* = \{\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_n\}$

for numbers of training iterations **do**

  Word-level representations $\mathbf{x}^w \leftarrow$ Equation (1)

  Character-level representations $\mathbf{x}^c \leftarrow$ Equation (2)

  Char_Global: $\mathbf{h}^{CG} \leftarrow f^{CG}(\mathbf{x}^c)$

  Char_Local: $\mathbf{h}^{CL} \leftarrow f^{CL}(\mathbf{x}^c)$

  Word_Global: $\mathbf{h}^{WG} \leftarrow f^{WG}(\mathbf{x}^w)$

  Word_Local: $\mathbf{h}^{WL} \leftarrow f^{WL}(\mathbf{x}^w)$

  Fused representation $\mathbf{h}_t$ based on above features by Equation (12)

  [*Contextual Feature Enhancing*]

  Additional contextual features $\mathbf{Z}^d_{i-1} \leftarrow$ Equation (14)

  Enhanced representation $\tilde{\mathbf{Z}}_i$ by Equation (15)

  $\hat{\mathbf{y}}^* \leftarrow$ *BiLSTM-CRF*$(\tilde{\mathbf{Z}}_i)$

end for

return Named entity labels $\hat{\mathbf{y}}^*$

Specifically, previous multiscale contextual features are aggregated with dilated convolutions, and are then cached and reused as an extended context for the next segment. Formally, let the two consecutive sequences be $\mathcal{S}_{i-1} = \{x_{i-1,1}, x_{i-1,2}, \ldots, x_{i-1,L}\}$ and $\mathcal{S}_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,L'}\}$, respectively. We denote the $(i-1)$th context representation as $\mathbf{H}_{i-1} = \{\mathbf{h}_{i-1,1}, \mathbf{h}_{i-1,2}, \ldots, \mathbf{h}_{i-1,t}, \ldots, \mathbf{h}_{i-1,L}\}(i > 1)$ based on (12). Thus, the current ith refined representation is obtained as follows:

$$\mathbf{Z}^d_{i-1} = \text{ReLU}(\text{Dconv}(\mathbf{H}_{i-1})) \tag{13}$$

$$\mathbf{Z}_{i-1} = \mathbf{Z}^{d=1}_{i-1} \oplus \mathbf{Z}^{d=2}_{i-1} \oplus \mathbf{Z}^{d=5}_{i-1} \tag{14}$$

$$\tilde{\mathbf{Z}}_i = \text{Cached}(\mathbf{Z}_{i-1}) \oplus \mathbf{Z}_i \tag{15}$$

where Dconv denotes the dilated CNN with a dilation rate $d \in \{1, 2, 5\}$, and the kernel size is set to 3. ReLU denotes the activation function, and Cached denotes fixing and caching without gradient operation. $\tilde{\mathbf{Z}}_i$ denotes the output refined representation incorporated with previous contextual features, which is then fed to the sequence labeling layer for final prediction (the upper part of Fig. 3).

Enhanced by the additional memory-distilled network, CAMFF explicitly uses fused contextual features in the history. Consequently, the effective context utilized for NER can exceed the current context. In general, CAMFF can cache previous context without limitation and reuse them as additional knowledge. However, in our experiments, we reused only consecutive previous contextual features owing to a limited GPU memory capacity.

E. Sequence Labeling for Final Prediction

The output refined representation with context-aware multilevel features is fed into a BiLSTM network to fully utilize all the semantic and syntactic information at a higher level. In addition, CRF enhances NER by considering neighboring labels to avoid mislabeling. For example, *I*-ORG cannot 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 that the current final representation output by 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 [31] is defined as $p(\hat{\mathbf{y}}|\mathbf{r}; \mathbf{W}, \mathbf{b})$ in CRF models as follows:

$$p(\hat{\mathbf{y}}|\mathbf{r}; \mathbf{W}, \mathbf{b}) = \frac{\prod_{t=1}^{n} \psi_t(\hat{y}_{t-1}, \hat{y}_t, \mathbf{r})}{\sum_{\hat{\mathbf{y}}' \in \mathcal{S}(\mathbf{r})} \prod_{t=1}^{n} \psi_t\left(\hat{y}'_{t-1}, \hat{y}'_t, \mathbf{r}\right)} \tag{16}$$

where $\hat{y}'$ represents a label chosen arbitrarily from all possible labels $\mathcal{S}(\mathbf{r})$, and $\psi_t(\hat{y}'_{t-1}, \hat{y}'_t, \mathbf{r}) = \exp(W_{\hat{y}'_{t-1}, \hat{y}'_t}\mathbf{r}_t + b_{\hat{y}'_{t-1}, \hat{y}'_t})$. Here $W_{\hat{y}'_{t-1}, \hat{y}'_t}$ and $b_{\hat{y}'_{t-1}, \hat{y}'_t}$ are the weight parameter and bias parameter corresponding to the label pair $(\hat{y}'_{t-1}, \hat{y}'_t)$.

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

$$L(\mathbf{W}, \mathbf{b}) = \sum_{r_t, \hat{\mathbf{y}}_t} \log(p(\hat{\mathbf{y}}_t|\mathbf{r}_t; \mathbf{W}, \mathbf{b})). \tag{17}$$

During the decoding phase, we search for the best label sequence $\hat{\mathbf{y}}^*$ that maximizes the likelihood as follows:

$$\hat{\mathbf{y}}^* = \arg\max_{\hat{y} \in \mathcal{S}(\mathbf{r})} p(\hat{\mathbf{y}}|\mathbf{r}; \mathbf{W}, \mathbf{b}). \tag{18}$$

Furthermore, for sequence labeling, we adopt a dynamic planning method named Viterbi to calculate the final tag sequence efficiently. Our complete training procedure for CAMFF is shown in Algorithm 1.

TABLE I

STATISTICS OF THESE FOUR DATASETS, #TOK DENOTES TOKENS AND #ENT DENOTES ENTITIES

Dataset		training	development	test	Types
CoNLL-2003	#tok	204,567	51,578	46,666	4
	#ent	23,499	5,942	5,648	
NCBI-disease	#tok	135,701	23,969	24,497	1
	#ent	5,134	787	960	
SciERC	#tok	45,762	6,571	13,501	6
	#ent	5,572	808	1,683	
JNLPBA	#tok	441,905	50,646	101,039	5
	#ent	46,390	4,911	8,662	

V. EXPERIMENTS

In this section, we first introduce the datasets and baseline methods for comparison. The experimental settings and metrics are then described in detail. Finally, the results and analyses are presented.

A. Datasets

To verify the effectiveness of the proposed frameworks, we conducted experiments on the CoNLL-2003 [36], NCBI-disease [37], SciERC [38], and JNLPBA [39] datasets,

TABLE II

EXPERIMENTAL RESULTS ON TEST SETS COMPARED TO THE CLASSIC AND STATE-OF-THE-ART METHODS. STANDARD PRECISION (P), RECALL (R), AND $F1$ SCORE ($F1$) ARE EMPLOYED AS EVALUATION METRICS

Model	CoNLL-2003			NCBI-disease			SciERC			JNLPBA		
	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
BiLSTM-CRF [5]	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 [35]	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 [29]	90.88	90.62	90.75	85.67	64.30	73.46	67.43	47.15	55.49	73.08	71.56	72.31
TreeBiLSTM [24]	90.22	90.44	90.33	79.26	82.83	81.00	61.29	56.69	58.90	69.56	68.95	69.25
DepBiLSTM [25]	90.56	90.85	90.70	82.00	77.28	79.57	59.11	56.34	57.69	69.16	70.74	69.94
CS Embeddings [7]	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 [9]	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 [18]	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-BI	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-NA	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
CAMFF	94.99	94.06	**94.53**	92.27	92.52	**92.39**	72.41	65.71	**68.89**	82.77	81.10	**81.93**

containing 4, 1, 6, and 5 entity types, respectively. All datasets were separated into training, development, and test sets. Table I presents the descriptive statistics of these four datasets.

1) CoNLL-2003 contains four types of named entities: PER, LOC, ORG, and MISC. It is a collection of newswire articles from the Reuters Corpus. The English version of the dataset was used in this work.
2) NCBI-disease contains a collection of 793 PubMed abstracts annotated at the mention and concept levels for disease name recognition, covering all the sentences in every PubMed citation.
3) SciERC is derived from 500 scientific abstracts. It includes annotations for scientific entities, their relations, and coreference clusters. The dataset contains six scientific entity types across 2687 sentences.
4) JNLPBA contains extract terms from molecular biology, such as PROTEIN, DNA, RNA, CELL_LINE, and CELL_TYPE. It was originally derived from the GENIA corpus. However, only the flat entities were preserved from the original corpus.

B. Baseline Methods

We compared our proposed models with the following baseline methods.

1) *BiLSTM-CRF [5]:* This applies the BiLSTM network to efficiently learn both past and future features of word embeddings, and uses a CRF layer to capture overall tag dependencies.
2) *BiLSTM-CNNs [35]:* 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.
3) *NeuralNER [29]:* Similar to Chiu and Nichols [35], this regards the word as a sequences of characters and learns character-level features from a BiLSTM, rather than CNNs.
4) *TreeBiLSTM [24]:* This introduces a tree-structured LSTM network, which is able to incorporate information from multiple child units.
5) *DepBiLSTM [25]:* This incorporates the features of long-distance and syntactic dependency graphs between words in a sentence to identify named entities.
6) *CS Embeddings [7]:* The neural language model generates context embeddings at the character level and obtains the final representation by concatenating pretrained word embeddings and character embeddings.
7) *SciBERT [9]:* It introduces a contextualized embedding model for scientific text based on BERT, which leverages unsupervised pretraining on a large corpus of publications and achieves the state-of-the-art on several tasks.
8) *CollaboNet [18]:* 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.

Among the above methods, BiLSTM-CRF, BiLSTM-CNNs, NeuralNER, TreeBiL-STM, and DepBiLSTM may be considered classic methods, whereas CS Embeddings, SciBERT, and CollaboNet are the state-of-the-art methods.

C. Experimental Settings and Evaluation Metrics

For experimental settings, we used both the pretrained word embeddings from GloVe and the randomly initialized character embeddings as input [40]. For word embeddings, the dimension was set to 300 and the word-level LSTM size was set to 250. For character embeddings, the dimension was set to 100, the character-level LSTM size was generally set to 25, and the CNN filter number was set to 50. For reusing contextual features, we adopted three multigrained dilated convolutions to iteratively incorporate the preceding previous context into the current context for NER, i.e., $L = 1$. The dilated rates were separately set to 1/2/5 with SAME padding, and the strides were set to 1. We train our proposed model using SGD to perform back-propagation through time. The batch size is set to 16. The learning rate was set to 0.001.

To avoid overfitting, dropout with a rate of 0.5 was applied to the input of each component as well as the output of the feature fusion layer in our model. For time efficiency, we generally initialized other related hyper-parameters according to the aforementioned baselines. We repeated the experiment 10 times with an early stopping strategy, and the average performance on the test set was reported as the final performance. All of our experiments were performed on the same machine with NVIDIA 2080ti GPU and Intel[1] Xeon E7-4870 CPU. We built our model with several widely-used libraries such as python, tensorflow-gpu, numpy, pathlib, etc.

For evaluation, we employed precision (P), recall (R), and $F1$-score ($F1$) as metrics and adopted the BIOES tagging scheme for all datasets. $F1$-score was chosen as the primary metric because it is the harmonic mean of precision and recall. Moreover, we designed the following four variants of the proposed model for ablation studies as follows.

1) *AMFF:* This is our basic model, incorporating multilevel features, e.g., character-level local features, character-level global features, word-level local features, and word-level global features, as shown in Fig. 3.
2) *AMFF-BI:* This incorporated a BiLSTM network into the WG component of the proposed AMFF.
3) *AMFF-NA:* This is a variant of the proposed AMFF without the attention mechanism.
4) *CAMFF:* This is the proposed AMFF enhanced with previous contextual features as illustrated in Fig. 4. Therefore, AMFF can be regarded to be a part of CAMFF.

D. Overall Results and Comparisons

Table II shows the experimental results for AMFF and its variants (AMFF-BI, AMFF-NA, and CAMFF), displaying as two groups, i.e., classic and state-of-the-art methods. For a fair comparison, we report their average results on the four benchmark datasets. Classic character-based and word-based NER methods obtained lower $F1$ scores than the recent methods on most benchmark datasets. In the case of the BiLSTM-CRF network, this might have occurred because of the bias caused by the previous label owing to insufficient information. Structured RNN baselines, i.e., TreeBiLSTM and DepBiLSTM, exhibited competitive performance on all datasets, demonstrating that NER could benefit from the dependency relations and long-distance dependencies. However, they performed slightly worse on CoNLL-2003 and JNLPBA than some other classic baselines, may be partly because of the quality of the dependency parsing.[2]

SciBERT achieved prior the state-of-the-art performances on NCBI-disease and SciERC, which depends on pretraining over a large corpus of scientific publications to generate contextualized embeddings. CollaboNet obtained the best result on JNLPBA because of multitask learning, which may make a wrong prediction when errors overlap. However, on CoNLL-2003, classic methods achieved $F1$ scores more than 90% and performed slightly better than SciBERT and CollaboNet. This is probably because these two methods were designed for academic and biomedical fields, respectively, and therefore failed to effectively capture general features such as local character features and lexical phrases. In addition, CS Embeddings achieved a prior state-of-the-art performance with 92.74% $F1$ score on CoNLL-2003 and also obtained competitive results on the other datasets, by merely taking the global word- and character-level features into consideration, which is partly similar to our proposed AMFF.

Overall, the state-of-the-art baselines generally outperformed the classic methods in terms of the $F1$ score. This is likely due to the richer information utilized by the recent methods, as shown in Table II. Consistent with this observation, our proposed method CAMFF achieved the best results on the four benchmark datasets, thereby validating its effectiveness. The mere incorporation of WG features based on attention enhanced the overall performance, as evidenced by AMFF-BI. This might be because the inter-word relations provided by pretrained word embeddings made BiLSTM redundant. Besides, there was a slight performance degradation when attention was omitted from the model (i.e., AMFF-NA), which highlights the importance of incorporating multilevel features based on attention mechanisms. CAMFF outperformed AMFF to achieve the best performance in terms of the $F1$ score on all four datasets. This indicates that broadening the scope of the current context considerably enhances NER. Further, CAMFF achieved the highest improvement of the $F1$ score on JNLPBA compared to the other three datasets. This may be due to the fact that the biomedical dataset contains more unusual entities than the other datasets, and therefore previous contextual features can provide supplementary knowledge for identifying these entities.

E. Ablation Study

To evaluate the contributions of the components (regarding WG, WL, CG, and CL) in AMFF and CAMFF, we conducted ablation experiments on the development sets of the four datasets. Table III shows that CAMFF and AMFF achieved overall better performance, demonstrating all components contribute to the recognition of named entities. Specifically, the performance of AMFF with only a single component was not satisfactory, whereas the fusion of multiple components made AMFF more competitive. This might be because the multilevel features captured from the four primary components of CG, CL, WG, and WL enhanced our model from multiple perspectives. Based on the attention mechanism, word-level components seemed to be more effective than character-level components, owing to the pretrained word embedding. On the SciERC dataset, the $F1$ score decreased to 65.86% when three components were fused. This could be due to the noise caused by the fusion of the components. However, in general, our proposed model tended to be more effective and robust when the number of components that were fused increased. This is mainly because each component contributed a unique perspective. CAMFF did not achieve the best performance on the development set of the NCBI-disease and SciERC datasets. This is perhaps due to noise from previous contexts. However,

[1]Registered trademark.
[2]https://spacy.io/

TABLE III

RESULTS OF THE ABLATION STUDY OF AMFF ON DEVELOPMENT SETS. "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

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
CAMFF_WG_CG_WL_CL	94.73	94.68	**94.70**	90.81	94.46	92.60	75.95	63.99	69.41	81.97	83.48	**82.76**

TABLE IV

OUR INDIVIDUAL ENTITY RESULTS ON FOUR DATASETS

Category	P(%)	R(%)	F1(%)
CoNLL-2003			
ORG	93.83	92.26	93.03
PER	97.75	98.06	97.91
LOC	96.23	96.23	96.23
MISC	91.12	92.18	91.63
macro avg	94.73	94.68	94.70
NCBI-disease			
Disease	90.81	94.46	92.60
macro avg	90.81	94.46	92.60
SciERC			
Method	82.25	75.91	78.95
OtherScientificTerm	80.43	67.03	73.12
Task	68.03	58.54	62.93
Generic	80.65	62.11	70.18
Material	73.40	61.07	66.67
Metric	70.94	59.29	64.59
macro avg	75.95	63.99	69.41
JNLPBA			
DNA	80.94	85.55	83.19
RNA	80.20	85.41	82.72
PROTEIN	80.60	86.92	83.64
CELL_TYPE	87.42	76.08	81.35
CELL_LINE	80.70	85.24	82.91
macro avg	81.97	83.84	82.76

CAMFF further improved the performance of NER on the CoNLL-2003 and JNLPBA datasets with $F1$ scores of 94.70% and 82.76%, respectively. This demonstrates its potential for enhancing NER and developing a more competitive model by leveraging a wider scope.

In addition, Table IV presents the performance of the CAMFF model for individual entity results on four datasets. The table shows that CAMFF exhibited competitive performances for fine-grained entity recognition. These results indicate that our CAMFF not only achieves superior performances for overall entity recognition, but also identifies individual entities accurately regarding $F1$ score across four datasets, further highlighting the superiority of our CAMFF.

F. Case Study for Detailed Analysis

Table V presents a case study comparing our model with CS Embeddings [7] and SciBERT [9], which are more representative than the others. In the example sentence, the word "Washington" is polysemous, i.e., the first "Washington" denotes an ORG together with "University," whereas the second in "George Washington" must be categorized as a PER. CS Embeddings and SciBERT may recognize "Washington" as either *B*-ORG or *B*-PER from the context, potentially causing "University" to be erroneously labeled *E*-PER because of the lack of auxiliary features.

TABLE V

CASE STUDY. THE BOLD WORDS ATTRACT MORE ATTENTION

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/CAMFF	**Washington**: [B-ORG]; University: [E-ORG] ; Missouri: [S-LOC]; George: [B-PER]; **Washington**: [E-PER]

In contrast to the existing methods, AMFF and CAMFF can easily recognize entities by considering additional features of the original sequences, such as the lexical phrases and the keyword "in," which are vital for distinguishing categorical entity labels. Furthermore, AMFF and CAMFF emphasize disambiguation based on self-attention, which enables the long-distance dependencies to be captured, as shown in Fig. 1. Our models are therefore advantageous for distinguishing polysemous words owing to their ability to fuse multilevel features from different perspectives.

G. Parameter Sensitivity Analysis

Four primary parameters, namely dropout rate, LSTM size, filter number, and batch size, were selected and their impacts on the effectiveness of AMFF were verified. The dropout rate denotes the percentage of units that are dropped in a neural network, the LSTM size controls the number of hidden state units used in sequence labeling, the filter number affects the output shape of the character-level CNN module, and the batch size controls training efficiency and the allocated

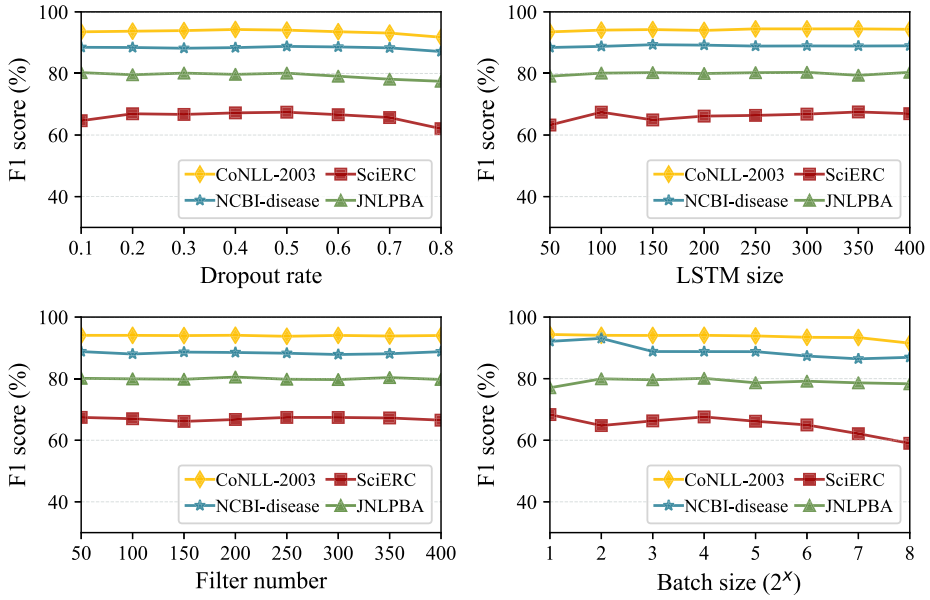

Fig. 5. Parameter sensitivity analysis for AMFF on these four datasets.

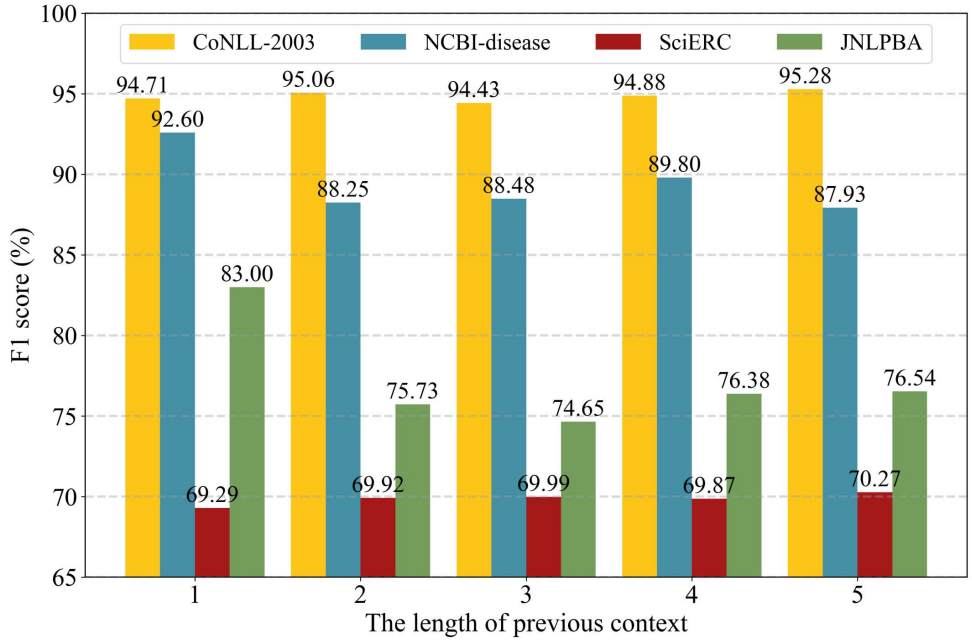

Fig. 6. Impact of the length of previous context for CAMFF. Different lengths of the previous context were incorporated, i.e., L, varied from 1 to 5, where 1 represents the preceding previous context, 2 represents the preceding two previous contexts, etc.

resources. To study the uncertainty in the output of our proposed model, we employed single-parameter sensitivity analysis by varying one parameter at a time while fixing the others each time. As shown in Fig. 5, AMFF maintained a high performance while the parameter varies on the four benchmark datasets. This demonstrates that multilevel features contribute to enhance NER, and further verifies the effectiveness and robustness of our proposed model.

H. Length of Previous Context

To explore the impact of the previous context, we conducted additional experiments on the development sets of these datasets by varying the length of the previous context while fixing other parameters. As shown in Fig. 6, different lengths of the previous context impacted the NER results differently. As a longer previous context did not always imply better NER, perhaps owing to data distribution and noise, we simply set the length of the previous context to 1. Overall, CAMFF maintained a high performance while the parameter varied, owing to the relevant contextual features obtained from a larger scope. This indicates that previous contextual features play an important role for NER in the current sequence. Therefore, CAMFF is effective and robust at dynamically utilizing context beyond current sequences.

I. Further Discussion

In this section, we will give a more in-depth analysis of the proposed frameworks in terms of effectiveness and robustness.

The proposed framework AMFF leverages the aggregation of relevant multilevel relevant features for NER. It outperformed a set of state-of-the-art baseline methods by a considerable margin. In CAMFF, which is a natural extension of the standard AMFF, previous contextual features demonstrated significant potentials to further enhance NER. CAMFF is therefore a novel solution for incorporating additional information, not only from different perspectives but also from a broader context, to effectively enhance NER. However, as CAMFF caches the previous contextual features, it inevitably requires more memory resources. To update the context, the memory slots used to store the previous context are updated once in one epoch to incorporate the preceding context. In the experiment, we set the length of the previous context to 1 to balance performance and resource consumption. In addition, both AMFF and CAMFF maintained a high performance while the model parameters (e.g., the dropout rate and the length of the previous context) varied as shown in Figs. 5 and 6, demonstrating that the frameworks are effective and robust. The frameworks can be easily extended and applied to enhance NER.

VI. Conclusion

This article presented a novel AMFF framework, i.e., AMFF, and a context-aware AMFF framework, i.e., CAMFF, which effectively leverage multilevel features for predicting categorical entity labels. The proposed frameworks capture character-level and word-level features from both global and local perspectives by adopting attention mechanisms. In CAMFF, we extended the scope of the current context by incorporating previous context and aggregating multiscale document-level features, which further improved the NER performance. Furthermore, the proposed frameworks can be easily extended by incorporating more discriminative features such as affixes to boost the NER performance. The experimental results demonstrated that CAMFF outperformed AMFF and various baseline methods, establishing new state-of-the-art results on the CoNLL-2003, NCBI-disease, SciERC, and JNLPBA datasets. Moreover, the analyses of the parameter sensitivity and the impact of the length of the previous context showed that our proposed frameworks are highly effective and robust.

For future work, the proposed frameworks, i.e., AMFF and CAMFF, are readily extended to improve the effectiveness in cross-language tasks, e.g., Chinese and English mixed corpus, and deal with downstream NLP applications, e.g., search engines, text mining systems, and fake news detection analysis.

References

[1] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 50–70, Jan. 2022.

[2] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12 pp. 2493–2537, Aug. 2011.

[3] K. R. Rahem and N. Omar, "Drug-related crime information extraction and analysis," in *Proc. 6th Int. Conf. Inf. Technol. Multimedia*, Nov. 2014, pp. 250–254.

[4] A. P. Quimbaya *et al.*, "Named entity recognition over electronic health records through a combined dictionary-based approach," *Proc. Comput. Sci.*, vol. 100, pp. 55–61, Jan. 2016.

[5] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015, *arXiv:1508.01991*.

[6] O. Kuru, O. A. Can, and D. Yuret, "Charner: Character-level named entity recognition," in *Proc. 26th Int. Conf. Comput. Linguistics*, 2016, pp. 911–921.

[7] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *Proc. 27th Int. Conf. Comput. linguistics*, 2018, pp. 1638–1649.

[8] Y. Xin, E. Hart, V. Mahajan, and J.-D. Ruvini, "Learning better internal structure of words for sequence labeling," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2584–2593.

[9] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3606–3611.

[10] Z. Yang, H. Chen, J. Zhang, J. Ma, and Y. Chang, "Attention-based multi-level feature fusion for named entity recognition," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, Jul. 2020, pp. 3594–3600.

[11] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[12] Y. Luo, F. Xiao, and H. Zhao, "Hierarchical contextualized representation for named entity recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 8441–8448.

[13] Z. Nasar, S. W. Jaffry, and M. K. Malik, "Named entity recognition and relation extraction: State-of-the-art," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–39, Apr. 2021.

[14] L. Liu *et al.*, "Empower sequence labeling with task-aware neural language model," in *Proc. AAAI Conf. Artif. Intell.*, vol. 2018, pp. 5253–5260.

[15] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 2741–2749.

[16] C. Dong, J. Zhang, C. Zong, M. Hattori, and H. Di, "Character-based LSTM-CRF with radical-level features for Chinese named entity recognition," in *Natural Language Understanding and Intelligent Applications*. Cham, Switzerland: Springer, 2016, pp. 239–250.

[17] T.-H. Pham and P. Le-Hong, "End-to-end recurrent neural network models for Vietnamese named entity recognition: Word-level vs. character-level," in *Proc. Int. Conf. Pacific Assoc. Comput. Linguistics*. Cham, Switzerland: Springer, 2017, pp. 219–232.

[18] W. Yoon, C. H. So, J. Lee, and J. Kang, "CollaboNet: Collaboration of deep neural networks for biomedical named entity recognition," *BMC Bioinf.*, vol. 20, no. S10, p. 249, May 2019.

[19] V. Yadav, R. Sharp, and S. Bethard, "Deep affix features improve neural named entity recognizers," in *Proc. 7th Joint Conf. Lexical Comput. Semantics*, 2018, pp. 167–172.

[20] C. Zheng, Y. Cai, J. Xu, H.-F. Leung, and G. Xu, "A boundary-aware neural model for nested named entity recognition," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 357–366.

[21] U. Khandelwal, H. He, P. Qi, and D. Jurafsky, "Sharp nearby, fuzzy far away: How neural language models use context," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2018, pp. 284–294.

[22] L. Luo *et al.*, "An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition," *Bioinformatics*, vol. 34, no. 8, pp. 1381–1388, 2017.

[23] A. Zukov-Gregoric, Y. Bachrach, P. Minkovsky, S. Coope, and B. Maksak, "Neural named entity recognition using a self-attention mechanism," in *Proc. IEEE 29th Int. Conf. Tools With Artif. Intell. (ICTAI)*, Nov. 2017, pp. 652–656.

[24] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. (Long Papers)*, vol. 1, 2015, pp. 1556–1566.

[25] Z. Jie and W. Lu, "Dependency-guided LSTM-CRF for named entity recognition," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3862–3872.

[26] J. Yu, B. Bohnet, and M. Poesio, "Named entity recognition as dependency parsing," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6470–6476.

[27] M. Xu, H. Jiang, and S. Watcharawittayakul, "A local detection approach for named entity recognition and mention detection," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2017, pp. 1237–1247.

[28] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 2978–2988.
[29] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," 2016, *arXiv:1603.01360*.
[30] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.
[31] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF," 2016, *arXiv:1603.01354*.
[32] Y. Liu, F. Meng, J. Zhang, J. Xu, Y. Chen, and J. Zhou, "GCDT: A global context enhanced deep transition architecture for sequence labeling," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 2431–2441.
[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[34] A. Miller, A. Fisch, J. Dodge, A.-H. Karimi, A. Bordes, and J. Weston, "Key-value memory networks for directly reading documents," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1400–1409.
[35] J. P. C. Chiu and E. Nichols, "Named entity recognition with bidirectional LSTM-CNNs," *Trans. Assoc. Comput. Linguistics*, vol. 4, pp. 357–370, Dec. 2016.
[36] E. F. T. K. Sang and F. D. Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," 2003, *arXiv:cs/0306050*.
[37] R. I. Doğan, R. Leaman, and Z. Lu, "NCBI disease corpus: A resource for disease name recognition and concept normalization," *J. Biomed. Inform.*, vol. 47, pp. 1–10, Feb. 2014.
[38] Y. Luan, L. He, M. Ostendorf, and H. Hajishirzi, "Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction," 2018, *arXiv:1808.09602*.
[39] J.-D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier, "Introduction to the bio-entity recognition task at JNLPBA," in *Proc. Int. Joint Workshop Natural Lang. Process. Biomed. Appl.*, Princeton, NJ, USA: Citeseer, 2004, pp. 70–75.
[40] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

Zhiwei Yang is currently pursuing the Ph.D. degree with the College of Computer Science and Technology, Jilin University (JLU), Changchun, China.

He has worked as a full-time Exchange Ph.D. Student with the Department of Computer Science, Hong Kong Baptist University (HKBU), Hong Kong, for one year, where he is currently a Research Assistant. His publications include International Joint Conferences on Artificial Intelligence (IJCAI), Conference on Empirical Methods in Natural Language Processing (EMNLP), IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (TNNLS), and *Neurocomputing*. His research interests include information extraction, rumor detection, and artificial intelligence.

Jing Ma received the Ph.D. degree from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2020.

She is currently an Assistant Professor with the Department of Computer Science, Hong Kong Baptist University (HKBU), Hong Kong. Her current research interests include natural language processing, information verification, and social media analytics.

Dr. Ma has been serving on the program committee for several international conferences, including the Annual Meeting of the Association for Computational Linguistics (ACL) and the Association for the Advancement of Artificial Intelligence (AAAI).

Hechang Chen received the Ph.D. degree from the College of Computer Science and Technology, Jilin University (JLU), Changchun, China, in December 2018. He was enrolled as a joint training Ph.D. Student at the University of Illinois at Chicago (UIC), Chicago, IL, USA, from November 2015 to December 2016, and a Visiting Student at Hong Kong Baptist University (HKBU), Hong Kong, from July 2017 to January 2018.

He is currently an Associate Professor with the School of Artificial Intelligence, JLU. He has published more than 40 articles in international journals and conferences, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI) and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (TNNLS). His current research interests lie in the areas of machine learning, data mining, complex network analysis, deep reinforcement learning, and knowledge graph.

Jiawei Zhang received the B.S. degree in computer science from Nanjing University, Nanjing, China, in 2012, and the Ph.D. degree in computer science from the University of Illinois at Chicago, Chicago, IL, USA, in 2017.

He founded the IFM Laboratory, University of California at Davis, Davis, CA, USA, in 2017, where he has been the Director since then. He is currently an Assistant Professor with the Department of Computer Science, University of California at Davis. Prior to joining UC Davis, he has worked with Florida State University, Tallahassee, FL, USA. He has published one textbook *Broad Learning through Fusions: An Application on Social Networks* and about 100 papers at top-tier academic conferences/journals in recent years.

Dr. Zhang, with his work, has received the "Best Student Paper Award (Runner-Up)" from Knowledge Discovery and Data Mining (KDD) 2018, and International Conference on Data Mining (ICDM) 2020. He serves as the PC/SPC for many top-tier conferences and a Panelist for NSF programs. He has organized the third HENA workshop at Conference on Information and Knowledge Management (CIKM) 2019. He was the Information Director and a Specialist of *ACM Transactions on Knowledge Discovery from Data* (TKDD) from 2014 to 2017.

Yi Chang (Senior Member, IEEE) is currently the Dean of the School of Artificial Intelligence, Jilin University (JLU), Changchun, China. He became a Chinese National Distinguished Professor in 2017 and the ACM Distinguished Scientist in 2018. Before joining academia, he was the Technical Vice President at Huawei Research America, in charge of knowledge graph, question answering, and vertical search projects. Before that, he was the Research Director of Yahoo Labs/Research, USA, from 2006 to 2016, in charge of search relevance of Yahoo's web search engine and vertical search engines. He is the author of two books and more than 100 papers in top conferences or journals. His research interests include information retrieval, data mining, machine learning, natural language processing, and artificial intelligence.

Dr. Chang won the Best Paper Award on ACM KDD 2016 and ACM International WSDM Conference 2016. He has served as one of the Conference General Chair for ACM International WSDM Conference 2018 and International ACM SIGIR Conference on Research and Development in Information Retrieval 2020. He is an Associate Editor of IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (TKDE).