Unified Named Entity Recognition as Word-Word Relation Classification

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, Fei Li
1 Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, China
2 Institute of Computing and Intelligence, Harbin Institute of Technology (Shenzhen), China
{theodorelee, hao.fei, liujiang, whuwshq, tengchong, dhji, lifei_csnlp}@whu.edu.cn, mason.zms@gmail.com

Abstract

So far, named entity recognition (NER) has been involved with three major types, including flat, overlapped (aka. nested), and discontinuous NER, which have mostly been studied individually. Recently, a growing interest has been built for unified NER, tackling the above three jobs concurrently with one single model. Current best-performing methods mainly include span-based and sequence-to-sequence models, where unfortunately the former merely focus on boundary identification and the latter may suffer from exposure bias. In this work, we present a novel alternative by modeling the unified NER as word-word relation classification, namely W²NER. The architecture resolves the kernel bottleneck of unified NER by effectively modeling the neighboring relations between entity words with Next-Neighboring-Word (NNW) and Tail-Head-Word-*(THW-*) relations. Based on the W²NER scheme we develop a neural framework, in which the unified NER is modeled as a 2D grid of word pairs. We then propose multi-granularity 2D convolutions for better refining the grid representations. Finally, a co-predictor is used to sufficiently reason the word-word relations. We perform extensive experiments on 14 widely-used benchmark datasets for flat, overlapped, and discontinuous NER (8 English and 6 Chinese datasets), where our model beats all the current top-performing baselines, pushing the state-of-the-art performances of unified NER.

Introduction

Named entity recognition (NER) has long been a fundamental task in natural language processing (NLP) community, due to its wide variety of knowledge-based applications, e.g., relation extraction (Wei et al. 2020; Li et al. 2021b), entity linking (Le and Titov 2018; Hou et al. 2020), etc. Studies of NER have gradually evolved initially from the flat NER (Lample et al. 2016; Strubell et al. 2017), late to the overlapped NER (Yu et al. 2020; Shen et al. 2021), and recently to the discontinuous NER (Dai et al. 2020; Li et al. 2021a). Specifically, flat NER simply detects the mention spans and their semantic categories from text, while the problems in overlapped and discontinuous NER become more complicated, i.e., overlapped entities contain the same tokens, and discontinuous entities entail non-adjacent spans, as illustrated in Figure 1.

Previous methods for multi-type NER can be roughly grouped into four major categories: 1) sequence labeling, 2) hypergraph-based methods, 3) sequence-to-sequence methods and 4) span-based methods. A majority of initial work formalizes NER as a sequence labeling problem (Lample et al. 2016; Zheng et al. 2019; Tang et al. 2018; Straková et al. 2019), assigning a tag to each token. However, it is difficult to design one tagging scheme for all NER subtasks. Then hypergraph-based models are proposed (Lu and Roth 2015; Wang and Lu 2018; Katiyar and Cardie 2018) to represent all entity spans, which however suffer from both the spurious structure and structural ambiguity issue during inference. Recently, Yan et al. (2021) propose a sequence-to-sequence (Seq2Seq) model to directly generate various entities, which unfortunately potentially suffers from the decoding efficiency problem and certain common shortages of Seq2Seq architecture, e.g., exposure bias. Span-based meth-

---

1Equal contribution
2Corresponding author
Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
ods (Luan et al. 2019; Li et al. 2021a) are another state-of-the-art (SoTA) approaches for unified NER, enumerating all possible spans and conduct span-level classification. Yet the span-based models can be subject to maximal span lengths and lead to considerable model complexity due to the enumerating nature. Thus, designing an effective unified NER system still remains challenging.

Most of the existing work has paid the major focus on how to accurately identify the entity boundary, i.e., the kernel problem of NER, especially for flat one (Straková et al. 2019; Fei et al. 2021). However, after carefully rethinking the common characteristics of all three types of NER, we find that the bottleneck of unified NER more lies in the modeling of the neighboring relations between entity words. Such adjacency correlations essentially describe the semantic connectivity between the partial text segments, which especially plays the key role for the overlapping and discontinuous ones. As exemplified in Figure 1(a), it could be effortless to detect the flat mention “aching in legs”, since its constituent words all are naturally adjacent. But, to detect out the discontinuous entity “aching in shoulders”, effectively capturing the semantic relations between the neighboring segments of “aching in” and “shoulders” is indispensable.

On the basis of the above observation, we in this paper investigate an alternative unified NER formalism with a novel word-word relation classification architecture, namely \( W^2 \)NER. Our method resolves the unified NER by effectively modeling both the entity boundary identification as well as the neighboring relations between entity words. Specifically, \( W^2 \)NER makes predictions for two types of relations, including the Next-Neighboring-Word (NNW) and the Tail-Head-Word-* (THW-*) , as illustrated in Figure 1(b). The NNW relation addresses entity word identification, indicating if two argument words are adjacent in an entity (e.g., aching→in), while the THW-* relation accounts for entity boundary and type detection, revealing if two argument words are the tail and head boundaries respectively of “*” entity (e.g., legs→aching, Symptom).

Based on the \( W^2 \)NER scheme, we further present a neural framework for unified NER (cf. Figure 3). First, BERT (Devlin et al. 2019) and BiLSTM (Lample et al. 2016) are used to provide contextualized word representations, based on which we construct a 2-dimensional (2D) grid for word pairs. Afterwards, we design multi-granularity 2D convolutions to refine the word-pair representations, effectively capturing the interactions between both the close and distant word pairs. A co-predictor finally reasons the word-word relations and produces all possible entity mentions, in which the biaffine and the multi-layer perceptron (MLP) classifiers are jointly employed for the complementary benefits.

We conduct extensive experiments on 14 datasets, ranging from 2 English and 4 Chinese datasets for flat NER, 3 English and 2 Chinese datasets for overlapped NER, 3 English datasets for discontinuous NER. Compared with 12 baselines for flat NER, 7 baselines for overlapped NER, 7 baselines for discontinuous NER, our model achieves the best performances on all the datasets, becoming the new SoTA method of unified NER. Our contributions include:

- We present an innovative method that casts unified NER as word-word relation classification, where both the relations between boundary-words and inside-words of entities are fully considered.
- We develop a neural framework for unified NER, in which we newly propose a multi-granularity 2D convolution method for sufficiently capturing the interactions between close and distant words.
- Our model pushes current SoTA performances of NER on total 14 datasets. Our code is available at https://github.com/ljynlp/W2NER.

**NER as Word-Word Relation Classification**

Flat, overlapped, discontinuous NER can be formalized as follows: given an input sentence consisting of \( N \) tokens or words \( X = \{ x_1, x_2, ..., x_N \} \), the task aims to extract the relations \( R \) between each token pairs \( (x_i, x_j) \), where \( R \) is pre-defined, including NONE, Next-Neighboring-Word (NNW), and Tail-Head-Word-* (THW-*). These relations can be explained as below and we also give an example as demonstrated in Figure 2 for better understanding.

- **NONE**, indicating that the word pair does not have any relation defined in this paper.
- **Next-Neighboring-Word**: the NNW relation indicates that the word pair belongs to an entity mention, and the word in certain row of the grid has a successive word in certain column of the grid.
- **Tail-Head-Word-***: the THW relation indicates that the word in certain row of the grid is the tail of an entity mention, and the word in certain column of the grid is the head of an entity mention. “*” indicates the entity type.

With such design, our framework is able to identify flat, overlapped and discontinuous entities simultaneously. As shown in Figure 2, it is effortless to decode out two
entities “aching in legs” and “aching in shoulders” by NNW relations (aching→in), (in→legs), and (in→shoulders), and THW relations (legs→aching, Symptom) and (shoulders→aching, Symptom). Moreover, NNW and THW relations imply other effects for NER. For example, NNW relations associate the segments of the same discontinuous entity (e.g., “aching in” and “shoulders”), and they are also beneficial for identifying entity words (neighbouring) and non-entity words (non-neighbouring). THW relations help identify the boundaries of entities, which plays an important role reported in recent NER studies (Zheng et al. 2019; Fei et al. 2021; Shen et al. 2021).

Unified NER Framework

The architecture of our framework is illustrated in Figure 3, which mainly consists of three components. First, the widely-used pretrained language model, BERT (Devlin et al. 2019), and bi-directional LSTM (Lample et al. 2016) are used as the encoder to yield contextualized word representations from input sentences. Then a convolution layer is used to build and refine the representation of the word-pair grid for later word-word relation classification. Afterward, a co-predictor layer (Li et al. 2021b) that contains a biaffine classifier and a multi-layer perceptron is leveraged for jointly reasoning the relations between all word pairs.

Encoder Layer

We leverage BERT (Devlin et al. 2019) as inputs for our model since it has been demonstrated to be one of the state-of-the-art models for representation learning in NER (Wang et al. 2021) and relation classification (Li et al. 2021b). Given an input sentence \( X = \{ x_1, x_2, ..., x_N \} \), we convert each token or word \( x_i \) into word pieces and then feed them into a pretrained BERT module. After the BERT calculation, each sentential word may involve vectorial representations of several pieces. Here we employ max pooling to produce word representations based on the word piece representations. To further enhance context modeling, we follow prior work (Wadden et al. 2019; Li et al. 2021a), adopting a bi-directional LSTM (Lample et al. 2016) to generate final word representations, i.e., \( H = \{ h_1, h_2, ..., h_N \} \in \mathbb{R}^{N \times d_h} \), where \( d_h \) denotes the dimension of a word representation.

Convolution Layer

We adopt convolution neural networks (CNNs) as the representation refiner, since CNNs are naturally suitable for 2-D convolution on the grid, and also show the very prominence on handling relation determination jobs (Zeng et al. 2014; Wang et al. 2016). Our convolution layer includes three modules, including a condition layer with normalization (Liu et al. 2021) for generating the representation of the word-pair grid, a BERT-style grid representation build-up to enrich the representation of the word-pair grid, and a multi-granularity dilated convolution for capturing the interactions between close and distant words.

Conditional Layer Normalization Since the goal of our framework is to predict the relations between word pairs, it is important to generate a high-quality representation of the word-pair grid, which can be regarded as a 3-dimensional matrix, \( V \in \mathbb{R}^{N \times N \times d_h} \), where \( V_{ij} \) denotes the representation of the word pair \( (x_i, x_j) \). Because both NNW and THW relations are directional, i.e., from a word \( x_i \) in certain row to a word \( x_j \) in certain column as shown in Figure 2 (e.g., aching→in and legs→aching), the representation \( V_{ij} \) of the word pair \( (x_i, x_j) \) can be considered as a combination of the representation \( h_i \) of \( x_i \) and \( h_j \) of \( x_j \), where the combination should imply that \( x_j \) is conditioned on \( x_i \). Inspired by Liu et al. (2021), we adopt the Conditional Layer Normalization (CLN) mechanism to calculate \( V_{ij} \):

\[
V_{ij} = \text{CLN}(h_i, h_j) = \gamma_{ij} \odot \left( \frac{h_j - \mu}{\sigma} \right) + \lambda_{ij},
\]

where \( h_i \) is the condition to generate the gain parameter \( \gamma_{ij} = W_{ij} h_i + b_\alpha \) and bias \( \lambda_{ij} = W_{\beta} h_i + b_\beta \) of layer normalization. \( \mu \) and \( \sigma \) are the mean and standard deviation across the elements of \( h_j \), denoted as:

\[
\mu = \frac{1}{d_h} \sum_{k=1}^{d_h} h_{jk}, \quad \sigma = \sqrt{\frac{1}{d_h} \sum_{k=1}^{d_h} (h_{jk} - \mu)^2}.
\]

where \( h_{jk} \) denotes the \( k \)-th dimension of \( h_j \).
BERT-Style Grid Representation Build-Up As everyone knows, the inputs of BERT (Devlin et al. 2019) consist of three parts, namely token embeddings, position embeddings and segment embeddings, which model word, position and sentential information respectively. Motivated by BERT, we enrich the representation of the word-pair grid using a similar idea, where the tensor \( \mathbf{V} \in \mathbb{R}^{N \times N \times d_v} \) represents word information, a tensor \( \mathbf{E}^d \in \mathbb{R}^{N \times N \times d_e} \) represents the relative position information between each pair of words, and a tensor \( \mathbf{E}^f \in \mathbb{R}^{N \times N \times d_f} \) represents the region information for distinguishing lower and upper triangle regions in the grid. We then concatenate three kinds of embeddings and adopt a multi-layer perceptron (MLP) to reduce their dimensions and mix these information to get the position-region-aware representation of the grid \( \mathbf{C} \in \mathbb{R}^{N \times N \times d_c} \). The overall process can be formulated as:

\[
\mathbf{C} = \text{MLP}_1 (\{[\mathbf{V}; \mathbf{E}^d; \mathbf{E}^f]\}).
\]

Multi-Granularity Dilated Convolution Motivated by TextCNN (Kim 2014), we adopt multiple 2-dimensional dilated convolutions (DConv) with different dilation rates \( l \) (e.g., \( l \in \{1, 2, 3\} \)) to capture the interactions between the words with different distances, because our model is to predict the relations between these words. The calculation in one dilated convolution can be formulated as:

\[
\mathbf{Q}^l = \sigma (\text{DConv}_l (\mathbf{C})),
\]

where \( \mathbf{Q}^l \in \mathbb{R}^{N \times N \times d_e} \) denotes the output of the dilation convolution with the dilation rate \( l \), \( \sigma \) is the GELU activation function (Hendrycks and Gimpel 2016). After that, we can obtain the final word-pair grid representation \( \mathbf{Q} = \{\mathbf{Q}^1, \mathbf{Q}^2, \mathbf{Q}^3\} \in \mathbb{R}^{N \times N \times 3d_e} \).

Co-Predictor Layer After the convolution layer, we obtain the word-pair grid representations \( \mathbf{Q} \), which are used to predict the relation between each pair of words using an MLP. However, prior work (Li et al. 2021b) has shown that MLP predictor can be enhanced by collaborating with a biaffine predictor for relation classification. We thus take these two predictors concurrently to calculate two separate relation distributions of word pairs \((x_i, x_j)\), and combine them as the final prediction.

Biaffine Predictor The input of the biaffine predictor is the output \( \mathbf{H} = \{h_1, h_2, \ldots, h_N\} \in \mathbb{R}^{N \times d_h} \) of the encoder layer, which can be considered as a residual connection (He et al. 2016) that is widely-used in current deep learning research. Given the word representations \( \mathbf{H} \), we use two MLPs to calculate the subject \((x_i)\) and object \((x_j)\) word representations, \( s_i \) and \( o_j \) respectively. Then, a biaffine classifier (Dozat and Manning 2017) is used to compute the relation scores between a pair of subject and object words \((x_i, x_j)\):

\[
s_i = \text{MLP}_2(h_i),
\]

\[
o_j = \text{MLP}_3(h_j),
\]

\[
y_{ij}' = s_i^\top U o_j + W[s_i; o_j] + b,
\]

where \( U, W \) and \( b \) are trainable parameters, \( s_i \) and \( o_j \) denote the subject and object representations of the \( i \)-th and \( j \)-th word, respectively. Here \( y_{ij}' \in \mathbb{R}^{\mathcal{R}} \) is the scores of the relations pre-defined in \( \mathcal{R} \).

MLP Predictor Based on the word-pair grid representation \( \mathbf{Q} \), we adopt an MLP to calculate relations scores for word pairs \((x_i, x_j)\) using \( \mathbf{Q}_{ij} \):

\[
y_{ij}'' = \text{MLP}(\mathbf{Q}_{ij}),
\]

where \( y_{ij}'' \in \mathbb{R}^{\mathcal{R}} \) is the scores of the relations pre-defined in \( \mathcal{R} \). The final relation probabilities \( y_{ij} \) for the word pair \((x_i, x_j)\) are calculated by combining the scores from the biaffine and MLP predictors:

\[
y_{ij} = \text{Softmax}(y_{ij}'+y_{ij}'').
\]

Decoding The predictions of our model are the words and their relations, which can be considered as a directional word graph. The decoding object is to find certain paths from one word to another word in the graph using NNW relations. Each path corresponds to an entity mention. Besides the type and boundary identification for NER, THW relations can also be used as auxiliary information for disambiguation. Figure 4 illustrates four cases for decoding from easy to difficult.

- In the example (a), two paths “A→B” and “D→E” correspond to flat entities, and THW relations indicate their boundaries and types.
- In the example (b), if there is no THW relation, we can only find one path and thus “BC” is missing. In contrast, with the help of THW relations, it is easy to identify that “BC” is nested in “ABC”, which demonstrates the necessity of THW relations.
- The case (c) shows how to identify discontinuous entities. Two paths “A→B→C” and “A→C→D” can be found, and the NNW relation contributes to connecting the discontinuous spans “AB” and “CD”.
- Considering a complex and rare case (d), it is impossible to decode correct entities “ACD” and “BCE” because we can find 4 paths in this ambiguous case using only NNW relations. In contrast, only using THW relations will recognize continuous entities (e.g., “ABCD”) rather than correct discontinuous entities (e.g., “ACD”). Therefore, we can obtain correct answers by collaboratively using both relations.

Learning For each sentence \( X = \{x_1, x_2, \ldots, x_N\} \), our training target is to minimize the negative log-likelihood losses with
Table 1: Results for English flat NER datasets. “†” denotes our re-implementation via their code. We run our model for 5 times and report averaged values.  

<table>
<thead>
<tr>
<th></th>
<th>CoNLL2003</th>
<th>OntoNotes 5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P  R  F1</td>
<td>P  R  F1</td>
</tr>
<tr>
<td><strong>Sequence Labeling</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lample et al. (2016)</td>
<td>-    -</td>
<td>90.94 - -</td>
</tr>
<tr>
<td>Strubell et al. (2017)</td>
<td>-    -</td>
<td>90.65 - -</td>
</tr>
<tr>
<td><strong>Span-based</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yu et al. (2020)</td>
<td>92.91</td>
<td>92.13 92.52 90.01 89.77 89.89</td>
</tr>
<tr>
<td>Shen et al. (2021)</td>
<td>92.13</td>
<td>93.73 92.94 - - -</td>
</tr>
<tr>
<td><strong>Hypergraph-based</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wang and Lu (2018)</td>
<td>-    -</td>
<td>90.50 - -</td>
</tr>
<tr>
<td><strong>Seq2Seq</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Straková et al. (2019)</td>
<td>-    -</td>
<td>92.98 - -</td>
</tr>
<tr>
<td>Yan et al. (2021)</td>
<td>92.56</td>
<td>93.73 92.94 - - -</td>
</tr>
<tr>
<td>W²NER (ours)</td>
<td>92.71</td>
<td>93.44 93.07 90.03 90.97 90.50</td>
</tr>
</tbody>
</table>

Table 2: Results for Chinese flat NER datasets. All the baselines are sequence labeling methods or their variations.  

<table>
<thead>
<tr>
<th></th>
<th>OntoNotes 4.0</th>
<th>MSRA</th>
<th>Resume</th>
<th>Weibo</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P  R  F1</td>
<td>P  R  F1</td>
<td>P  R  F1</td>
<td>P  R  F1</td>
</tr>
<tr>
<td>Zhang and Yang (2018)</td>
<td>76.35 71.56 73.88</td>
<td>93.57 92.79 93.18</td>
<td>94.81 94.11 94.46</td>
<td>53.04 62.25 58.79</td>
</tr>
<tr>
<td>Yan et al. (2019)</td>
<td>-    -</td>
<td>72.43 - -</td>
<td>-    -</td>
<td>-    -</td>
</tr>
<tr>
<td>Gui et al. (2019)</td>
<td>76.40 72.60 74.45</td>
<td>94.50 92.93 93.71</td>
<td>95.37 94.84 95.11</td>
<td>57.14 66.67 59.92</td>
</tr>
<tr>
<td>Li et al. (2020b)</td>
<td>-    -</td>
<td>81.82 - -</td>
<td>-    -</td>
<td>-    -</td>
</tr>
<tr>
<td>Ma et al. (2020)</td>
<td>83.41 82.21 82.81</td>
<td>95.75 95.10 95.42</td>
<td>96.08 96.13 96.11</td>
<td>70.94 67.02 70.50</td>
</tr>
<tr>
<td>W²NER (ours)</td>
<td>82.31 83.36 83.08</td>
<td>96.12 96.08 96.10</td>
<td>96.96 96.35 96.65</td>
<td>70.84 73.87 72.32</td>
</tr>
</tbody>
</table>

Table 3: Results for English overlapped NER datasets.  

<table>
<thead>
<tr>
<th></th>
<th>ACE2004</th>
<th>ACE2005</th>
<th>GENIA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P  R  F1</td>
<td>P  R  F1</td>
<td>P  R  F1</td>
</tr>
<tr>
<td><strong>Sequence Labeling</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ju et al. (2018)</td>
<td>-    -</td>
<td>74.20 70.30 72.20</td>
<td>78.50 71.30 74.70</td>
</tr>
<tr>
<td>Wang et al. (2020)</td>
<td>86.08 86.48 86.28</td>
<td>83.95 85.39 84.66</td>
<td>79.45 78.94 79.19</td>
</tr>
<tr>
<td><strong>Span-based</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yu et al. (2020)</td>
<td>87.30 86.00 86.70</td>
<td>85.20 85.60 85.40</td>
<td>81.80 79.30 80.50</td>
</tr>
<tr>
<td>Shen et al. (2021)</td>
<td>87.44 87.38 87.41</td>
<td>86.09 87.27 86.67</td>
<td>80.19 80.89 80.54</td>
</tr>
<tr>
<td><strong>Hypergraph-based</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wang and Lu (2018)</td>
<td>78.00 72.40 75.10</td>
<td>76.80 72.30 74.50</td>
<td>77.00 73.30 75.10</td>
</tr>
<tr>
<td><strong>Seq2Seq</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Straková et al. (2019)</td>
<td>-    -</td>
<td>84.33 - -</td>
<td>-    -</td>
</tr>
<tr>
<td>Yan et al. (2021)</td>
<td>87.27 86.41 86.84</td>
<td>83.16 86.38 84.74</td>
<td>78.87 79.60 79.23</td>
</tr>
<tr>
<td>W²NER (ours)</td>
<td>87.33 87.71 87.52</td>
<td>85.03 88.62 86.79</td>
<td>83.10 79.76 81.39</td>
</tr>
</tbody>
</table>

regards to the corresponding gold labels, formalized as:
\[
\mathcal{L} = -\frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{r=1}^{|\mathcal{R}|} \hat{y}_{ij}^r \log y_{ij}^r, \tag{10}
\]
where \(N\) is the number of words in the sentence, \(\hat{y}_{ij}^r\) is the binary vector that denotes the gold relation labels for the word pair \((x_i, x_j)\), and \(y_{ij}^r\) are the predicted probability vector. \(r\) indicates the \(r\)-th relation of the pre-defined relation set \(\mathcal{R}\).  

**Experimental Settings**

**Datasets**

To evaluate our framework for three NER subtasks, we conducted experiments on 14 datasets.

**Flat NER Datasets** We adopt CoNLL-2003 (Sang and Meulder 2003) and OntoNotes 5.0 (Pradhan et al. 2013b) in English, OntoNotes 4.0 (Weischedel et al. 2011), MSRA (Levow 2006), Weibo (Peng and Dredze 2015; He and Sun 2017), and Resume (Zhang and Yang 2018) in Chinese. We employ the same experimental settings in previous work (Lample et al. 2016; Yan et al. 2021; Ma et al. 2020; Li et al. 2020b).

**Overlapped NER Datasets** We conduct experiments on ACE 2004 (Doddington et al. 2004), ACE 2005 (Walker et al. 2011), GENIA (Kim et al. 2003). For GENIA, we follow Yan et al. (2021) to use five types of entities and split the train/dev/test as 8:1:0:9:1:0. For ACE 2004 and ACE 2005 in English, we use the same data split as Lu and Roth (2015); Yu et al. (2020). For ACE 2004 and ACE 2005 in Chinese, we split the train/dev/test as 8:1:0:9:1:0.

**Discontinuous NER Datasets** We experiment on three datasets for discontinuous NER, namely CADEC (Karimi et al. 2015), ShARe13 (Pradhan et al. 2013a) and ShARe14 (Mowery et al. 2014), all of which are derived from biomedical or clinical domain documents. We use the preprocessing scripts provided by Dai et al. (2020) for data splitting.
Table 4: Results for discontinuous NER datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>CADEC P</th>
<th>CADEC R</th>
<th>CADEC F1</th>
<th>ShAre13 P</th>
<th>ShAre13 R</th>
<th>ShAre13 F1</th>
<th>ShAre14 P</th>
<th>ShAre14 R</th>
<th>ShAre14 F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tang et al. (2018)</td>
<td>67.80</td>
<td>64.99</td>
<td>66.36</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Li et al. (2021a)</td>
<td>72.10</td>
<td>48.40</td>
<td>58.00</td>
<td>83.80</td>
<td>60.40</td>
<td>70.30</td>
<td>79.10</td>
<td>70.70</td>
<td>74.70</td>
</tr>
<tr>
<td>Yan et al. (2021)</td>
<td>70.08</td>
<td>71.21</td>
<td>70.64</td>
<td>82.09</td>
<td>77.42</td>
<td>79.69</td>
<td>77.20</td>
<td>83.75</td>
<td>80.34</td>
</tr>
<tr>
<td>Dai et al. (2020)</td>
<td>68.90</td>
<td>69.00</td>
<td>69.00</td>
<td>80.50</td>
<td>75.00</td>
<td>77.70</td>
<td>78.10</td>
<td>84.70</td>
<td>81.30</td>
</tr>
<tr>
<td>Wang et al. (2021)</td>
<td>70.50</td>
<td>72.50</td>
<td>71.50</td>
<td>84.30</td>
<td>78.20</td>
<td>81.20</td>
<td>78.20</td>
<td>84.70</td>
<td>81.30</td>
</tr>
<tr>
<td>W2NER (ours)</td>
<td>74.09</td>
<td>72.35</td>
<td>73.21</td>
<td>85.57</td>
<td>79.68</td>
<td>82.52</td>
<td>79.88</td>
<td>83.71</td>
<td>81.75</td>
</tr>
</tbody>
</table>

Table 5: F1s for Chinese overlapped NER datasets. Models with “×” are adapted to target datasets using their code.

<table>
<thead>
<tr>
<th>Method</th>
<th>ACE2004 F1</th>
<th>ACE2005 F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yu et al. (2020) ×</td>
<td>87.35</td>
<td>88.39</td>
</tr>
<tr>
<td>Shen et al. (2021) ×</td>
<td>87.47</td>
<td>88.21</td>
</tr>
<tr>
<td>W2NER (ours)</td>
<td>88.00</td>
<td>88.81</td>
</tr>
</tbody>
</table>

Figure 5: Results of overlapped (a) and discontinuous mentions (b) on ShAre14.
Table 6: Model ablation studies (F1s). DConv(l=1) denotes the convolution with the dilation rate 1.

Model Ablation Studies

We ablate each part of our model on the CoNLL2003, ACE2005 and CADEC datasets, as shown in Table 6. First, without region and distance embeddings, we observe slight performance drops on the three datasets. By removing all convolutions, the performance also drops obviously, which verifies the usefulness of the multi-granularity dilated convolution. Furthermore, after removing convolutions with different dilation rate, the performance also decreases, especially for the convolution with the dilation rate 2.

Comparing the biaffine and MLP in the co-predictor layer, we find that although the MLP plays a leading role, the bi-affine also brings about 1% gains on average. When both are removed, the performance drops on the three datasets. By removing the MLP, we observe slight performance drops on the ACE2005 and CADEC datasets.

Related Work on NER

Sequencing Labeling Approaches NER is usually considered as a sequence labeling problem, to assign each token a tag from a pre-designed tagging scheme (e.g., BIO). Current mainstream work combine the CRF (Lafferty et al. 2001; Finkel et al. 2005) with neural architecture, such as CNN (Collobert et al. 2011; Strubell et al. 2017), bi-directional LSTM (Huang et al. 2015; Lample et al. 2016), and Transformer (Yan et al. 2019; Lu et al. 2020b). However, these methods fail to directly solve the problem of overlapping or discontinuous NER. Ju et al. (2018) propose a neural model for nested NER by dynamically stacking flat NER layers. Tang et al. (2018) extend the BIO label scheme to BIOHD to address the problem of discontinuous mention.

Span-based Approaches There have been several studies that cast NER as span-level classification, i.e., enumerating all possible spans, and determining if they are valid mentions and the types (Xu et al. 2017; Luan et al. 2019; Yamada et al. 2020). Yu et al. (2020b) utilize biaffine attention (Drozat and Manning 2017) to measure the possibility as a mention of a text span. Li et al. (2020a) reformulate NER as a machine reading comprehension (MRC) task and extract entities as the answer spans. Shen et al. (2021) implement a two-stage identifier to generate span proposals through a filter and a regressor, and then classify them into the corresponding categories. Li et al. (2021a) convert the discontinuous NER to find complete subgraphs from a span-based entity fragment graph, and achieve competitive results. But, due to the exhaustively enumerating nature, those methods suffer from maximal span lengths and considerable model complexity, especially for long-span entities.


Sequence-to-Sequence Approaches Gillick et al. (2016) first apply the Seq2Seq model for NER, taking as inputs the sentence, and outputting all the entity start positions, span lengths and labels. Straková et al. (2019) use the Seq2Seq architecture for overlapping NER with enhanced BILOU scheme. Fei et al. (2021) employ Seq2Seq with pointer network for discontinuous NER. The latest attempt in (Yan et al. 2021) tackles the unified NER via a Seq2Seq model with pointer network based-on BART (Lewis et al. 2020), generating a sequence of all possible entity start-end indexes and types. Seq2Seq architecture unfortunately suffers from the potential decoding efficiency problem as well as the exposure bias issue.

Differences between Our Approach and Previous Approaches Most of the existing NER work mainly consider more accurate entity boundary identification. In this work, we explore a different task modeling for unified NER, i.e., a formalism as word-word relation classification. Our method can effectively model the relations between both the boundary-words and inside-words of entities. Also, our method with 2D grid-tagging can substantially avoid the drawbacks in current best-performing baselines, e.g., span-based and sequence-to-sequence models.

Conclusion

In this paper, we propose a novel unified NER framework based on word-word relation classification to address unified NER concurrently. The relations between word pairs are predefined as next-neighboring-word relations and tail-head-word relations. We find that our framework is quite effective for various NER, which achieves SoTA performances for 14 widely-used benchmark datasets. Moreover, we propose a novel backbone model that consists of a BERT-BiLSTM encoder layer, a convolution layer for building and refining the representation of the word-pair grid, and a co-predictor layer for jointly reasoning relations. Through ablation studies, we find that our convolution-centric model performs well and several proposed modules such as the co-predictor and grid representation enrichment are also effective. Our framework and model are easy to follow, which will promote the development of NER research.
Acknowledgments
This work is supported by the National Natural Science Foundation of China (No.61772378, No. 62176187), the National Key Research and Development Program of China (No. 2017YFC1200500), the Research Foundation of Ministry of Education of China (No.18JZD015), the Youth Fund for Humanities and Social Science Research of Ministry of Education of China (No. 22YJCZH064). This work is also the research result of the independent scientific research project (humanities and social sciences) of Wuhan University, supported by the Fundamental Research Funds for the Central Universities.

References


