Pretraining Multi-modal Representations for Chinese NER Task with Cross-Modality Attention

Chengcheng Mai
Nanjing University
Nanjing, Jiangsu Province, China
maicc@smail.nju.edu.cn

Mengchuan Qiu
Nanjing University
Nanjing, Jiangsu Province, China
mengchuan.qiu@smail.nju.edu.cn

Kaiwen Luo
Nanjing University
Nanjing, Jiangsu Province, China
kaiwenluo@smail.nju.edu.cn

Ziyan Peng
Nanjing University
Nanjing, Jiangsu Province, China
pengziyan@smail.nju.edu.cn

Jian Liu
Nanjing University
Nanjing, Jiangsu Province, China
brooksj@foxmail.com

Chunfeng Yuan
Nanjing University
Nanjing, Jiangsu Province, China
cfyuan@nju.edu.cn

Yihua Huang*
Nanjing University
Nanjing, Jiangsu Province, China
yhuang@nju.edu.cn

ABSTRACT

Named Entity Recognition (NER) aims to identify the pre-defined entities from the unstructured text. Compared with English NER, Chinese NER faces more challenges: the ambiguity problem in entity boundary recognition due to unavailable explicit delimiters between Chinese characters, and the out-of-vocabulary (OOV) problem caused by rare Chinese characters. However, two important features specific to the Chinese language are ignored by previous studies: glyphs and phonetics, which contain rich semantic information of Chinese. To overcome these issues by exploiting the linguistic potential of Chinese as a logographic language, we present **MPM-CNER** (short for **M**ulti-modal **P**retraining **M**odel for **C**hinese **NER**), a model for learning multi-modal representations of Chinese semantics, glyphs, and phonetics, via four pre-training tasks: Radical Consistency Identification (RCI), Glyph Image Classification (GIC), Phonetic Consistency Identification (PCI), and Phonetic Classification Modeling (PCM). Meanwhile, a novel cross-modality attention mechanism is proposed to fuse these multi-modal features for further improvement. The experimental results show that our method outperforms the state-of-the-art baseline methods on four benchmark datasets, and the ablation study also verifies the effectiveness of the pre-trained multi-modal representations.

CCS CONCEPTS

• **Computing methodologies → Information extraction**.

*Corresponding author.

WSDM '22, February 21–25, 2022, Tempe, AZ, USA.

ACM ISBN 978-1-4503-9132-0/22/02…$15.00
https://doi.org/10.1145/3488560.3498450

KEYWORDS

Chinese named entity recognition, multi-modal representations, pre-training model, cross-modality attention

ACM Reference Format:
Chengcheng Mai, Mengchuan Qiu, Kaiwen Luo, Ziyan Peng, Jian Liu, Chunfeng Yuan, and Yihua Huang. 2022. Pretraining Multi-modal Representations for Chinese NER Task with Cross-Modality Attention. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining (WSDM '22), February 21–25, 2022, Tempe, AZ, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3488560.3498450

1 INTRODUCTION

As a fundamental task of identifying structured entities from unstructured text, Named Entity Recognition (NER) plays a crucial role in many Natural Language Processing (NLP) downstream applications, such as representation learning [10, 28], knowledge graph [18, 43], and Information retrieval [6], etc.

Despite the ever-evolving deep learning architectures that have witnessed great success in performing NER on English corpus [37, 38], Chinese NER still faces some language-specific difficulties, like the ambiguity of entity boundary recognition caused by the absence of the explicit delimiters between Chinese characters [20, 23]. To overcome these obstacles, some variant architectures of neural networks based on LSTM (short for Long Short-Term Memory) [11] were proposed. Zhang and Yang [41] presented a lattice-based structure to identify the entity boundaries on the character level. Recently, introducing external knowledge, like lexicon information, into the deep learning models also attracted the attention of researchers. Ma et al. [25] incorporated all the possible entity words into the Chinese NER model and indicated the relative position for each character in the words. Li et al. [16] converted the NER task into the Machine Reading Comprehension (MRC) task, which takes advantage of the semantic prior information from the comprehensive descriptions of entity categories.

However, previous studies mainly focus on the contextual semantic information of the input Chinese sentences but ignore the

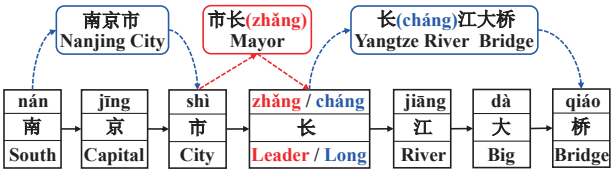

Figure 1: An example of the polyphonic phenomenon in entity boundary recognition. The Latin letters above the rectangle box are the Chinese Pinyin, representing the pronunciation of Chinese characters. Character '长' has two pronunciations, corresponding to two different entity segmentations. The blue rounded rectangles and the dashed arrows denote the correctly identified entities and the entity boundaries, while the red ones denote the incorrect entities.

unique linguistic features of Chinese as a logographic language [36], i.e., the phonetic and glyph features.

The existence of polyphonic characters is a common phenomenon in Chinese phonetics [34]. The different pronunciations of a polyphonic Chinese character correspond to different semantic meanings, which prompts the recognition of entity boundaries. As observed in Figure 1, the Chinese character '长' has two pronunciations, symbolized by Pinyin [42], the official Latinized pronunciation system for Chinese characters, as 'cháng [$\text{tʂ}^{\text{h}}\text{a}\eta^{35}$]' and 'zhǎng [$\text{tʂa}\eta^{214}$]'. When '长' is pronounced as 'cháng', it means the 'length', which helps to correctly identify the entity boundary and the entity '长(cháng)江大桥', while means the 'leader' when pronounced as 'zhǎng', resulting in the wrong identification of '市长(zhǎng)' as an entity. This phenomenon shows that developing the modality of Chinese phonetics has great potential for improving Chinese NER.

Apart from the phonetic features, as a pictographic language, Chinese characters also contain rich glyphic information that is highly related to the semantic modality [39]. The Chinese character is composed of multiple radicals. The neighboring characters in a Chinese word or an entity sharing the same radical are likely to have similar semantics [36]. For example, characters representing geographic locations like '河流 (river)', '湖泊 (lake)', and '海洋 (ocean)' have the same radical '氵(water)'. Another benefit of the radical is to alleviate the out-of-vocabulary (OOV) problem because the meanings of rare Chinese characters can be inferred from the shared radicals and the context. For instance, although character '澎' in the word '澎湖 (pescadores)' is an uncommon Chinese character, the meaning of this word still can be guessed to be related to a geographical entity, like a lake, according to the radical '氵(water)' and the neighboring character '湖 (lake)'.

It is a non-trivial task to learn the generic representations of Chinese NER from the above multi-modal features. The existing multi-modal pre-training models mainly focus on the fields of image-text generation [32], video question answering [15], and visually grounded dialog [4, 22], etc., which also brings trouble, that is, these models need a large amount of data from multiple modalities to align the image and text data.

In contrast, our method takes another route by fully exploiting the semantics, phonetics, and glyphs information contained in the Chinese language itself, and hardly require extra aligned corpora. First, for the semantic information, we adopted the BERT [5] model to encode the Chinese characters; for the glyph information, we utilized the Swin-Transformer [21], which serves as a backbone for encoding visual features, to encode the glyph embedding from the images of Chinese characters with pre-training tasks: Radical Consistency Identification (RCI) and Glyph Image Classification (GIC). Similarly, for the phonetic information, we converted the pronunciation audio of Chinese characters into the Mel-spectrogram [27] and encoded the phonetic feature based on the Swin-Transformer with pre-training tasks: Phonetic Consistency Identification (PCI) and Phonetic Classification Modeling (PCM). Then, to dynamically evaluate the contributions of different modality features in varying contexts, a novel cross-modality attention mechanism was proposed to fuse the multi-modal representations and establish the interaction between different modalities, for learning the representations of Chinese characters. Finally, the multi-modal representations of characters were input into the Conditional Random Field (CRF) [14] based decoder for extracting Chinese entities.

It is worth noting that our MPM-CNER has an additional advantage in dealing with smaller datasets of Chinese NER, which benefits from the full exploitation of the Chinese language. The detailed results will be given in the ablation experiment part. To the best of our knowledge, we are the first to combine the multi-modal features to the pre-training model for Chinese NER, which has been rarely explored before.

The contributions of our work are summarized as follows:

- A novel multi-modal pre-training model is proposed based on the Transformer architecture with four pre-training tasks to achieve the generic representations from Chinese semantics, phonetics, and glyphs, for Chinese NER.
- We present a cross-modality attention mechanism to fuse the multi-modal features by dynamically evaluating their contributions to the performance of Chinese NER.
- The extensive experimental results show that our method outperforms the existing state-of-the-art baseline models across four popular benchmark datasets and verify the effectiveness of the pre-training paradigm along with the cross-modality attention.

2 OUR MPM-CNER METHOD

The overview of our MPM-CNER method is shown in Figure 2.

2.1 Input Embeddings

For the Chinese NER task, the input sentence s is denoted as the character sequence: $s = \{c_1, \ldots, c_i, \ldots, c_n\}$. The output of sentence s is the predicted entity label sequence, denoted as $L = \{l_1, \ldots, l_i, \ldots, l_n\}$ in the form of tagging scheme 'B/M/E/O/S-Type', where 'B' and 'E' represent the beginning and end character of an entity, 'M' signifies that character c_i is inside an entity, 'S' means the single-word entity, 'O' means that c_i does not belong to an entity, and 'Type' represents the entity type.

In our method, the input character is firstly converted into three feature embeddings corresponding to different modalities: semantic embedding, glyph embedding, and phonetic embedding.

Semantic Embedding. Considering that BERT, as one kind of Transformer [33], has pushed forward the state-of-the-art models

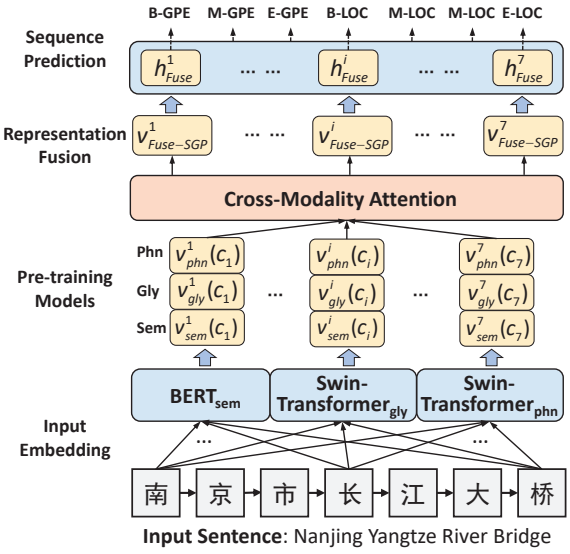

Figure 2: The overview of the proposed MPM-CNER model. The first annotation tag 'B-GPE' represents the relative position, 'Beginning', and entity type, 'GPE', for entity '南京市(Nanjing City)'.

for a large range of NLP tasks, we adopted BERT to encode each character in the sentence to obtain the semantic embedding of character c_i. We initialized the semantic embedding for character c_i by summing up the corresponding character and position embeddings, denoted as:

$$v^i_{sem} = \mathbf{e}_{sem}(c_i) \tag{1}$$

where $\mathbf{e}_{sem}(\cdot)$ represents the semantic embedding lookup table learned by BERT.

Glyph Embedding. To obtain the glyph embedding of each character, we first converted the input Simplified Chinese character into Traditional Chinese character, for retaining more associations between glyphs and semantics, and retrieved the corresponding glyph image of character c_i, denoted as $gly(c_i)$, from Xinhua Dictionary[1], which is the most authoritative Chinese dictionary. Then, we used the Swin-Transformer [21], which has an iterative structure of L layers, to encode the glyph embedding of $gly(c_i)$:

$$\begin{aligned}
&v^l_{gly}(c_i) = LinearEmebed\left(PatchPart\left(gly\left(c_i\right)\right)\right),\ l=1\\
&\begin{cases}
\hat{v}^l_{gly}(c_i) = W\text{-}MSA\left(LN\left(v^{l-1}_{gly}(c_i)\right)\right) + v^{l-1}_{gly}(c_i),\\
v^l_{gly}(c_i) = MLP\left(LN\left(\hat{v}^l_{gly}(c_i)\right)\right) + \hat{v}^l_{gly}(c_i), \quad 1 < l \le L\\
\hat{v}^{l+1}_{gly}(c_i) = SW\text{-}MSA\left(LN\left(v^l_{gly}(c_i)\right)\right) + v^l_{gly}(c_i),\\
v^{l+1}_{gly}(c_i) = MLP\left(LN\left(\hat{v}^{l+1}_{gly}(c_i)\right)\right) + \hat{v}^{l+1}_{gly}(c_i),
\end{cases}
\end{aligned} \tag{2}$$

where $PatchPart$ means to partition the glyph image into multiple non-overlapping patches and $LinearEmbed$ was then applied on these patches to obtain the dense glyph vectors. LN and MLP represent the LayerNorm and Multi-Layer Perception. $W\text{-}MSA$ and $SW\text{-}MSA$ denote the window and shifted window-based multi-head self-attention module, respectively. The patches were merged along the layers, following the original Swin-Transformer. We used the output of the last layer, $v^L_{gly}(c_i)$, as the glyph embedding of character c_i, denoted as:

$$v^i_{gly} = v^L_{gly}(c_i) \tag{3}$$

Phonetic Embedding. To obtain the phonetic embedding of the input character, we used the Pypinyin Library[2] to annotate each input character with the Chinese Pinyin, based on the input sentence. Then, we constructed an audio database containing 1,310 kinds of pronunciations for almost all the Chinese characters and found the corresponding pronunciation audio according to the Pinyin of the Chinese character, denoted as $audio(c_i)$.

To extract the phonetic features from the pronunciation audio, we converted the $audio(c_i)$ into the Mel-spectrogram, denoted as $Melspec(c_i)$, which is a two-dimensional graph that reflects the strength and frequency of the pronunciation audio, using the Librosa[3] tool. The final phonetic embedding of character c_i is:

$$v^i_{phn}(c_i) = Swin\text{-}Transformer(Melspec(c_i)) \tag{4}$$

where the detailed calculation process in $Swin\text{-}Transformer(\cdot)$ is the same as that in Equation 2.

Then, these embedding features were put into the pre-training tasks for multi-modal representation learning.

2.2 Pre-training Tasks for Multi-modal Representations

We proposed four pre-training tasks as follows. The first two tasks were designed for the glyph modality, while the last two were for the phonetic modality.

Radical Consistency Identification (RCI). The goal of the RCI task is to judge whether the radicals of two input Chinese character images are the same, so as to obtain radical-sensitive glyph representations. As illustrated in Figure 3, we first constructed a glyph database containing 18,467 Traditional Chinese characters, named **GlyDB**. Next, we randomly selected a Chinese character from **GlyDB** and then chose another Chinese character with the same radical with 50% probability, or a character with a different radical with a probability of 50%. Then, the glyph embeddings of the two selected Chinese characters, denoted as $v^i_{gly}(c_i)$ and $v^j_{gly}(c_j)$, were input into the $Swin\text{-}Transformer_{gly}$. Lastly, we concatenated the glyph embeddings of the two Chinese characters learned in the last layer of the $Swin\text{-}Transformer_{gly}$ model and fed it into a Fully Connected (FC) layer followed by a sigmoid function to predict whether the radicals of the two Chinese characters are the same ('Yes' or 'No'). The binary loss function is defined as:

$$L_{RCI} = -\frac{1}{N_{gly}}\sum_{i=1}^{N_{gly}} y^i_{RCI}\log(p^i_{RCI}) + (1-y^i_{RCI})\log(1-p^i_{RCI}) \tag{5}$$

where $P^i_{RCI} = sigmoid(\mathbf{W}_{RCI}[v^i_{gly}(c_i); v^j_{gly}(c_j)] + b_{RCI})$, representing the probability that the radicals of the two characters are the same and y^i_{RCI} is the truth label. ';' denotes the concatenation operation. $\mathbf{W_{RCI}}$ and b_{RCI} are the trainable parameters, and N_{gly}

[1] https://www.cp.com.cn/service/download.html

[2] https://pypi.python.org/pypi/pypinyin

[3] http://librosa.org/doc/latest/index.html#librosa

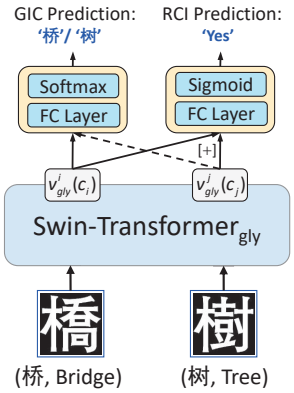

Figure 3: Pre-training tasks for glyph modality. The characters in brackets are the simplified version of the Traditional Chinese characters shown in the gray-scale glyph images. [+] represents the concatenation operation. Because characters '桥 (bridge)' and '树 (tree)' have the same radical, the prediction result of the RCI task was 'Yes'. The GIC task also predicted the correct Chinese characters based on the glyphs.

indicates the total number of sampled characters from **GlyDB** with six different fonts, (i.e., N_{gly}=110,802).

Glyph Image Classification (GIC). GIC is essentially a multi-class classification task, designed to identify the correct Chinese characters corresponding to the input glyphs. As shown in Figure 3, we input the glyph embeddings of the two Chinese characters into an FC layer followed by a softmax function, respectively, to predict the probability distribution of the corresponding characters. The multi-class loss function is given as:

$$L_{GIC} = -\frac{1}{N_{gly}} \sum_{i=1}^{N_{gly}} \sum_{k=1}^{K_{GIC}} y_{GIC}^{i,k} \log(p_{GIC}^{i,k}) \tag{6}$$

where $P_{GIC}^{i,k} = softmax(\mathbf{W}_{GIC} v_{gly}^{i}(c_i) + b_{GIC})$, representing the probability that $gly(c_i)$ corresponds to character c_k, and $y_{GIC}^{i,k}$ denotes the truth label of the character. $\mathbf{W_{GIC}}$ and b_{GIC} are the trainable parameters, K_{GIC} is the number of Chinese characters in **GlyDB**, (i.e., K_{GIC}=18,467).

Considering that task RCI and GIC are designed to share the same $Swin\text{-}Transformer_{gly}$ architecture, we applied the joint loss function for optimizing the parameters, denoted as:

$$L_{gly} = \lambda_{gly}^{1} L_{RCI} + \lambda_{gly}^{2} L_{GIC} \tag{7}$$

where both hyper-parameter λ_{gly}^{1} and λ_{gly}^{2} were set to 0.5.

Phonetic Consistency Identification (PCI). The Chinese pronunciation system consists of three parts: the initial, final and tone, among which the tones can be subdivided into five types: the high-level tone (¯), rising tone (´), falling-rising tone (ˇ), falling tone (ˋ), and soft tone ( ) [24]. The PCI task aims to determine whether the initials and finals of the two Chinese characters are consistent, so as to refine the pronunciation characteristics of the Chinese language.

To perform the PCI task, firstly, we build an audio database containing almost all the Chinese character pronunciations, named **PhnDB**, which contains 1,310 kinds of pronunciations of Chinese characters. Next, we randomly selected a pronunciation audio from **PhnDB**, and then randomly selected another pronunciation audio with the same initial and final as the first pronunciation audio with a probability of 50%, or selected a pronunciation audio that is different from the initial or final of the first audio, with a 50% chance. We converted the two pronunciation audios into the Mel-spectrograms and extracted the phonetic embeddings from them, which were then input into the $Swin\text{-}Transformer_{phn}$ model for pre-training. Lastly, the two phonetic embeddings, $v_{phn}^{i}(c_i)$ and $v_{phn}^{j}(c_j)$, learned in the last layer of the $Swin\text{-}Transformer_{phn}$ model were concatenated and input into an FC layer followed by a sigmoid function to predict whether the initials and finals of the two audios are both the same ('Yes' or 'No'). For example, in Figure 4, the finals of pronunciation 'cháng' and 'zhǎng' are the same, but the initials are different, so the classifier determined that the two characters are pronounced differently. The objective function to optimize the $Swin\text{-}Transformer_{phn}$ model is denoted as:

$$L_{PCI} = -\frac{1}{N_{phn}} \sum_{i=1}^{N_{phn}} y_{PCI}^{i} \log(p_{PCI}^{i}) + (1 - y_{PCI}^{i}) \log(1 - p_{PCI}^{i}) \tag{8}$$

where $P_{PCI}^{i} = sigmoid(\mathbf{W}_{PCI}[v_{phn}^{i}(c_i); v_{phn}^{j}(c_j)] + b_{PCI})$, representing the probability that the initials and finals of the two pronunciation audios are the same and y_{PCI}^{i} indicates the truth label. $\mathbf{W}_{PCI}$ and b_{PCI} are the trainable parameters, and N_{phn} denotes the total number of sampled audios from the **PhnDB** database, (i.e., N_{phn}=2,620, doubled by flipping the Mel-spectrograms for data augmentation).

Phonetic Classification Modeling (PCM). As illustrated in Figure 4, to learn the initial, final, and tone features of Chinese pronunciations, we input the phonetic embeddings of Chinese characters, trained by the $Swin\text{-}Transformer_{phn}$ model, into an FC layer followed by a softmax function, and then classified the input pronunciation audio into its corresponding Chinese pronunciation category. The optimization objective of the PCM task is:

$$L_{PCM} = -\frac{1}{N_{phn}} \sum_{i=1}^{N_{phn}} \sum_{k=1}^{K_{PCM}} y_{PCM}^{i,k} \log(p_{PCM}^{i,k}) \tag{9}$$

where $P_{PCM}^{i,k} = softmax(\mathbf{W}_{PCM} v_{phn}^{i}(c_i) + b_{PCM})$, representing the probability that pronunciation $audio(c_i)$ belongs to the pinyin of character c_i, and $y^{i,k}$ denotes the truth label of the input audio. $\mathbf{W}_{PCM}$ and b_{PCM} are the trainable parameters, K_{PCM} is the number of pronunciation audios in **PhnDB**, (i.e., K_{PCM}=1,310).

Similarly, we jointly optimized the loss function by minimizing the negative log-likelihood:

$$L_{phn} = \lambda_{phn}^{1} L_{PCI} + \lambda_{phn}^{2} L_{PCM} \tag{10}$$

where both hyper-parameter λ_{phn}^{1} and λ_{phn}^{2} were set to 0.5.

2.3 Cross-Modality Attention for Representation Fusion

After obtaining the multi-modal representations for the input character c_i, denoted as $v_{sem}^{i}(c_i)$, $v_{gly}^{i}(c_i)$, and $v_{phn}^{i}(c_i)$, respectively, through the above four pre-training tasks, we further proposed a

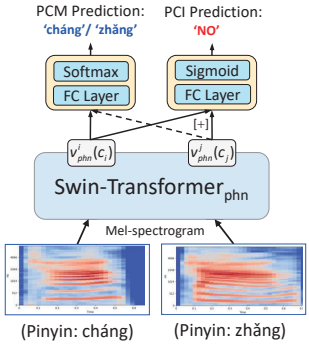

Figure 4: The pre-training tasks for the phonetic modality. [+] represents the vector concatenation operation. The inputs are the Mel-spectrograms of the pronunciation audios.

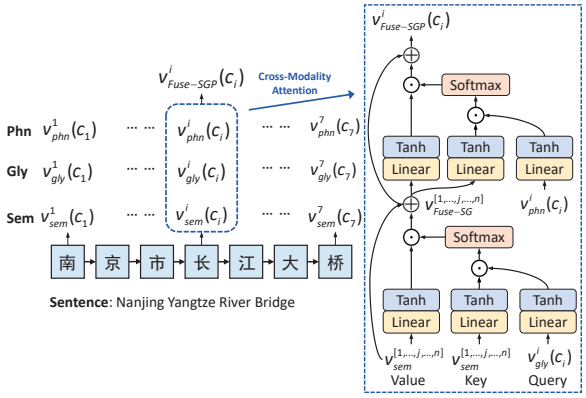

Figure 5: The illustration of the cross-modality attention mechanism for multi-modal representation fusion. ⊙ and ⊕ represent the vector multiplication and addition operations, respectively.

novel cross-modality attention mechanism to fuse these representations by evaluate the contributions of different modalities.

As illustrated in Figure 5, firstly, for the glyph representation sequence $V_{gly} = \{v^1_{gly}(c_1), \ldots, v^i_{gly}(c_i), \ldots, v^n_{gly}(c_n)\}$, $(1 \le i \le n)$, we combined the semantic representation with the glyph representation, calculated as:

$$\begin{aligned} \hat{v}^i_{gly}(c_i) &= W_{g\to s}\, v^i_{gly}(c_i) \\ \alpha^{i,j}_{Fuse-SG} &= \frac{\exp\left(s\left(v^j_{sem}(c_j), \hat{v}^i_{gly}(c_i)\right)\right)}{\sum_{k=1}^{n}\exp\left(s\left(v^k_{sem}(c_k), \hat{v}^i_{gly}(c_i)\right)\right)} \\ s\left(v^j_{sem}(c_j), \hat{v}^i_{gly}(c_i)\right) &= \frac{v^j_{sem}(c_j)^T\, \hat{v}^i_{gly}(c_i)}{\sqrt{D}} \\ Atten(v^{[1,\ldots,j,\ldots,n]}_{sem}, v^i_{gly}(c_i)) &= \sum_{j=1}^{n} \alpha^{i,j}_{Fuse-SG} \cdot v^j_{sem}(c_j) \end{aligned} \tag{11}$$

where $W_{g\to s}$ represents the transformation matrix that converts the dimension of $v^i_{gly}(c_i)$ to the same as $v^i_{sem}(c_i)$. D represents the dimension of the semantic and the converted glyph representation vectors. The fused representation after combing the semantic and glyph representations was denoted as:

$$v^i_{Fuse\text{-}SG}(c_i) = \text{Atten}\left(v^{[1,\ldots,j,\ldots n]}_{sem}, v^i_{gly}(c_i)\right) + v^i_{sem}(c_i) \tag{12}$$

Then, for the phonetic representation sequence $V_{phn} = \{v^1_{phn}(c_1), \ldots, v^i_{phn}(c_i), \ldots v^n_{phn}(c_n)\}$, we iteratively integrated the phonetic representation $v^i_{phn}(c_i)$ with $v^i_{Fuse\text{-}SG}(c_i)$ in the same way as in Equation 11 and Equation 12, calculated as:

$$v^i_{Fuse\text{-}SGP}(c_i) = \text{Atten}\left(v^{[1,\ldots,j,\ldots,n]}_{Fuse\text{-}SG}, v^i_{phn}(c_i)\right) + v^i_{Fuse\text{-}SG}(c_i) \tag{13}$$

Finally, we achieved the multi-modal representations for Chinese characters.

2.4 Sequence Prediction Layer

After acquiring the fused embedding of the multi-modal representations for character c_i, we input $v^i_{Fuse\text{-}SGP}(c_i)$ into the Bi-LSTM model [8] to further aggregate the contextual information from its neighboring characters. The final output hidden state is $h^i_{Fuse} = Bi\text{-}LSTM\left(v^i_{Fuse\text{-}SGP}(c_i)\right)$.

Afterwards, the standard CRF model was adopted to predict the probability of the entity label sequence $y = \{l_i\}|^n_{i=1}$, annotated with the tagging scheme in the form of 'B/M/E/O/S-Type'. The formula was as follows:

$$p(y \mid s) = \frac{\exp\left(\sum_{i=1}^{n} \mathbf{W}^{l_i}_{CRF} h^i_{Fuse} + b^{l_{i-1},l_i}_{CRF}\right)}{\sum_{y'\in Y(s)}\exp\left(\sum_{i=1}^{n} \mathbf{W}^{l'_i}_{CRF} h^i_{Fuse} + b^{l'_{i-1},l'_i}_{CRF}\right)} \tag{14}$$

where $Y(s)$ represents all the possible label sequences of sentence s and y' denotes one of the label sequences in $Y(s)$. $\mathbf{W}^*_{\mathbf{CRF}}$ and b^*_{CRF} are the trainable parameters. During decoding, we utilized the Viterbi algorithm to find the optimal label sequence with the highest probability. Given N labeled training data $\{(s_i, y_i)\}|^N_{i=1}$, we minimize the negative log-likelihood to train our model:

$$Loss = -\sum_{i=1}^{N} \log\left(p\left(y_i \mid s_i\right)\right) \tag{15}$$

3 EXPERIMENTS AND ANALYSES

3.1 Experiment Setup

Datasets and Metrics. Four popular Chinese NER datasets were adopted to evaluate our proposed method, including OntoNotes4.0 [35], MSRA [40], Resume [41], and Weibo [26]. OntoNotes4.0 and MSRA were collected from the news. Resume consisted of the resume data from the website Sina[4]. Weibo was crawled from social media[5]. The span-level micro-averaged Precision (P), Recall (R), and the F1 score were adopted as the evaluation metrics.

In addition, we collected 18,467 traditional Chinese characters as the dataset, named **GlyDB**, for pre-training tasks on the glyph modality, and 1,310 kinds of pronunciations for almost all the

[4]http://finance.sina.com.cn/stock/index.shtml
[5]https://www.weibo.com/

Table 1: Results on OntoNotes4.0 dataset

Models	P	R	F1
BERT+LSTM+CRF [25]	81.99	81.65	81.82
LR-CNN [9]	76.40	72.60	74.45
WC-LSTM [20]	76.09	72.85	74.43
Lattice-LSTM [41]	76.35	71.56	73.88
FLAT [17]	-	-	76.45
FLAT+BERT [17]	-	-	81.82
MRC+BERT [16]	82.98	81.25	82.11
PLTE+BERT [23]	79.62	81.82	80.60
SoftLexicon+BERT [25]	83.41	82.21	82.81
Glyce+BERT [36]	81.87	81.40	81.63
ChineseBERT [31]	80.77	**83.65**	82.18
LEBERT [19]	-	-	82.08
MPM-CNER(ours)	**84.30**	80.33	**83.21**

Table 2: Results on MSRA dataset

Models	P	R	F1
BERT+LSTM+CRF [25]	95.06	94.61	94.83
LR-CNN [9]	94.50	92.93	93.71
WC-LSTM [20]	94.58	92.91	93.74
Lattice-LSTM [41]	93.57	92.79	93.18
FLAT [17]	-	-	94.12
FLAT+BERT [17]	-	-	96.09
MRC+BERT [16]	96.18	95.12	95.75
PLTE+BERT [23]	94.91	94.15	94.53
SoftLexicon+BERT [25]	95.75	95.10	95.42
Glyce+BERT [36]	95.57	**95.51**	95.54
LEBERT [19]	-	-	95.70
MPM-CNER(ours)	**97.12**	95.34	**96.22**

Table 3: Results on Resume dataset

Models	P	R	F1
BERT+LSTM+CRF [25]	95.75	95.28	95.51
LR-CNN [9]	95.37	94.84	95.11
WC-LSTM [20]	95.27	95.15	95.21
Lattice LSTM [41]	94.81	94.11	94.46
FLAT [17]	-	-	95.45
FLAT+BERT [17]	-	-	95.86
PLTE+BERT [23]	96.16	**96.75**	96.45
SoftLexicon+BERT [25]	96.08	96.13	96.11
Glyce+BERT [36]	96.62	96.48	96.54
LEBERT [19]	-	-	96.08
MPM-CNER(ours)	**97.18**	96.03	**96.60**

Chinese characters as the audio dataset, named **PhnDB**, for pre-training tasks on the phonetic modality.

Implementation Details. In this paper, the data split of the training, validation, and test set followed the previous work [25]. The dimensions of the hidden state for BERT were set to 768 and 144 for Swin-Transformer. Adam [13] was adopted to optimize our model, with initial learning rates of 3e-5 for BERT and 1e-4 for Swin-Transformer. The batch size was set to 24 on all four datasets. The dropout rates for OntoNote4.0, MSRA, Resume, Weibo were 0.3, 0.3, 0.1, and 0.5, respectively. The training epochs were set to 5 for dataset OntoNote4.0 and MSRA, and 10 for Resume and Weibo. The patch size in Swin-Transformer was set to 6×6. The sizes of the gray-scale glyph images and Mel-spectrograms of pronunciation audios were 48×48×1 and 48×48×3, respectively. Other parameters were the same as those in the original Swin-Transformer paper [21]. The experiments were conducted on one NVIDIA Tesla T4 GPU.

3.2 Effectiveness Study

As shown in Table 1 to 4, our MPM-CNER method outperforms all the compared baseline methods across four benchmark datasets for Chinese NER, which verifies the effectiveness of our method.

For **OntoNotes4.0** dataset, our method surpasses the second-best method, SoftLexicon+BERT [25], by +0.4 on F1 score. In addition, our MPM-CNER method outperforms method Glyce+BERT, which also utilizes the glyph features of Chinese characters, by a large margin of +1.58 in terms of F1 score. One possible reason is that Glyce+BERT does not take into account the phonetic features and fuse the multi-modal representations.

For **MSRA** dataset, our method outperforms the second-best model, FLAT+BERT, by +0.13 on F1 score. For **Resume** dataset, our method also achieves the new state-of-the-art (SOTA) result compared to the second-best model, Glyce+BERT, with an improvement of +0.06 in terms of F1 score.

For **Weibo** dataset, compared with SoftLexicon+BERT [25], which does not introduce the multi-modal information, our method achieves a huge improvement of +1.54 on F1 score. One possible explanation is that the Weibo dataset is crawled from social media, and the content published by users is more casual. Therefore, the introduction of multi-modal information can play a role in data augmentation. For the second-best method, ChineseBERT, which also uses the glyph and pinyin features, our method still outperforms it by +1.24 on F1 score, which may be attributed to our use of the pronunciation audio of Chinese characters and our proposed cross-modality attention.

3.3 Ablation Study

Influences of model components. As shown in Table 5 from line 1 to 3, after removing the phonetics ('-w/o Phn'), glyphs ('-w/o Gly'), and phonetics+glyphs ('-w/o Phn+Gly') modality from our full model, the F1 scores on four datasets also decreased gradually. The performance degradation verifies the necessity of multi-modal features for our method. Besides, we found that when the multi-modal components were removed on Weibo dataset, the F1 score decreased significantly by 3.13, which suggests that multi-modal features are more crucial for social media. We also removed the cross-modality attention ('-w/o CM-Atten') and directly concatenated the multi-modal representations of input characters. As a result, the F1 scores on four datasets decreased, which verifies the necessity of the cross-modality attention.

Furthermore, we removed the pre-training phase ('-w/o Pre-training') and found that the F1 scores on four datasets all decreased, indicating the importance of the pre-training model for boosting the

Table 4: Results on Weibo dataset

Models	P	R	F1
Peng and Dredze [26]	66.47	47.22	55.28
Cao et al. [1]	55.72	50.68	53.08
Ding et al. [7]	63.10	56.30	59.50
BERT+LSTM+CRF [25]	-	-	67.33
LR-CNN [9]	-	-	59.92
WC-LSTM [20]	-	-	59.84
Lattice-LSTM [41]	-	-	58.79
FLAT [17]	-	-	63.42
FLAT+BERT [17]	-	-	68.55
PLTE+BERT [23]	72.00	66.67	69.23
SoftLexicon+BERT [25]	-	-	70.50
Glyce+BERT [36]	67.68	67.71	67.60
ChineseBERT [31]	68.75	**72.97**	70.80
LEBERT [19]	-	-	70.75
MPM-CNER (ours)	**72.98**	71.12	**72.04**

performance of our method. Once again, we observed that the F1 score on Weibo dataset dropped the most (-3.61 on F1 score), which shows that the pre-training strategy can be used as an effective data augmentation method for social media.

Influences of data size. To demonstrate the effectiveness of our MPM-CNER method in low data resource settings, we randomly selected the subsets of 10%, 20%, and 50% from the training datasets to train our method and the baseline models. In the following experiments, we have tried our best to re-implement method FLAT+BERT and BERT+LSTM+CRF. According to Table 6, we observed that: **1)** our method outperforms the two baseline methods across various sizes of training data. **2)** The F1 scores grow monotonically when the data size increases. These phenomena indicate that the more data we have, the better the Chinese NER methods perform. However, if there are fewer data available, which is quite common in practical application scenarios, our MPM-CNER method has advantages over other Chinese NER methods in dealing with a small amount of data.

Influences of polyphonic characters. Polyphonic character is a common phenomenon in the Chinese language. To evaluate the influences of the polyphonic characters, we further subdivided each test set into two parts: one contains polyphonic characters, while another does not. According to the experimental results listed in Table 7, we observed that, for datasets MSRA, Resume, and Weibo, our method and the baseline methods all achieved better performance on the data subsets without polyphonic characters, compared with the subsets containing polyphonic characters. For dataset OntoNotes, methods performed on the data sets containing polyphonic characters achieved better results. In addition, our MPM-CNER method outperformed the baseline methods in most cases, whether the datasets contain polyphonic characters or not. Therefore, the phenomenon of polyphony in Chinese plays an important role in Chinese NER.

Influences of out-of-vocabulary (OOV) characters. To evaluate the influences of the OOV characters, which only appear in the test set of the dataset, not in the vocabulary of BERT, we split the test set into two parts: containing OOV characters or not. The experiments were performed on the two parts and the results were given in Table 8. We found that **1)** our method surpasses other baseline methods, whether the dataset contains OOV characters or not. **2)** all the methods achieve better results on the dataset that does not contain the OOV characters, which means the OOV characters

Table 5: Ablation experiment results on four datasets. F1 scores are reported. '-w/o' means to remove one certain component from our MPM-CNER method, e.g., '-w/o Phn' means to remove the phonetic modality from our model.

Model variants	OntoNotes	MSRA	Resume	Weibo
MPM-CNER (ours)	**83.21**	**96.22**	**96.60**	**72.04**
-w/o Phn	82.74 (-0.47)	96.14 (-0.08)	96.30 (-0.30)	70.35 (-1.69)
-w/o Gly	82.48 (-0.73)	95.98 (-0.24)	96.18 (-0.42)	68.99 (-3.05)
-w/o Phn+Gly	81.18 (-2.03)	95.76 (-0.46)	95.78 (-0.82)	68.91 (-3.13)
-w/o CM-Atten	83.11 (-0.10)	95.75 (-0.47)	96.38 (-0.22)	70.31 (-1.73)
-w/o Pre-training	80.30 (-2.91)	96.01 (-0.21)	95.83 (-0.77)	68.43 (-3.61)

Table 6: Performance as a function of the percentage of the training data used during the training process. F1 scores are reported on four datasets.

Data Size	Methods	OntoNotes	MSRA	Resume	Weibo
10%	FLAT+BERT	67.90	88.06	89.62	55.93
	BERT+LSTM+CRF	78.60	92.87	94.49	61.29
	MPM-CNER (ours)	**79.53**	**93.12**	**94.98**	**64.63**
20%	FLAT+BERT	73.53	91.22	93.29	64.68
	BERT+LSTM+CRF	80.02	93.74	94.90	65.66
	MPM-CNER (ours)	**80.11**	**93.86**	**95.46**	**66.67**
50%	FLAT+BERT	78.58	93.65	93.87	65.93
	BERT+LSTM+CRF	81.21	94.64	95.37	66.34
	MPM-CNER (ours)	**81.98**	**95.48**	**95.98**	**68.40**
100%	FLAT+BERT [17]	81.82	96.09	95.86	68.55
	BERT+LSTM+CRF [25]	81.82	94.83	95.51	68.43
	MPM-CNER (ours)	**83.21**	**96.22**	**96.60**	**72.04**

Table 7: F1 scores are reported to evaluate the influences of polyphonic characters on four datasets. '-w/o polyPhn' means the subset of data that does not contain polyphonic characters. '-w polyPhn' means the subset of data that contains polyphonic characters.

Dataset		FLAT+BERT	BERT+ LSTM+CRF	MPM-CNER (ours)
OntoNotes	-w/o polyPhn	78.53	80.52	**82.73**
	-w polyPhn	79.83	81.90	**84.65**
MSRA	-w/o polyPhn	94.84	95.51	**96.33**
	-w polyPhn	93.25	95.10	**96.16**
Resume	-w/o polyPhn	97.03	**97.80**	97.78
	-w polyPhn	94.79	94.97	**96.29**
Weibo	-w/o polyPhn	69.76	69.57	**73.28**
	-w polyPhn	57.14	70.05	**70.81**

Table 8: F1 scores are reported to evaluate the influences of OOV characters on four datasets. '-w/o OOV' means the subset of data that does not contain OOV characters. '-w OOV' means the subset that contains OOV characters.

Dataset		FLAT+BERT	BERT+ LSTM+CRF	MPM-CNER (ours)
OntoNotes	-w/o OOV	79.45	81.36	**83.88**
	-w OOV	53.19	78.16	**82.98**
MSRA	-w/o OOV	94.10	95.32	**96.27**
	-w OOV	65.45	88.89	**89.55**
Resume	-w/o OOV	95.45	95.81	**96.78**
	-w OOV	66.67	62.50	**71.43**
Weibo	-w/o OOV	63.24	69.67	**71.89**
	-w OOV	81.37	82.84	**85.71**

lead to the performance degradation for Chinese NER. Only the Weibo dataset is an exception. One possible reason is that the size of Weibo dataset is small, and only eight entities in the test set contain OOV characters, so its results are not representative.

3.4 Case Study

An example is illustrated in Table 9 to intuitively demonstrate the effectiveness of introducing the semantic, glyph, and phonetic features for Chinese NER. In this example, our method can correctly identify the entity '中国 (China)', while other two baseline methods wrongly concatenated the characters '中国 (China)' and '镍 (nickel)' and recognized them as an incorrect entity '中国镍', which means 'the nickel from China'. We attribute the success of our method to the prompt of Chinese radicals from the glyph information. Considering that Chinese character '镍 (nickel)' and '钴 (cobalt)' share the same radical '钅', which means the 'metal', and there is a comma between them, the two characters are likely to have similar meanings and form a coordinative phrase. Therefore, it can be inferred that Chinese characters '中国 (China)' and '镍 (nickel)' should not be concatenated and recognized as an entity.

Table 9: An example from OntoNotes4.0 dataset. Characters colored in blue and red represent the correct and incorrect recognized entity, respectively. 'GPE' means the entity type. 'Gold seg' and 'Ground Truth' represent the correct word segmentations and human-annotated entities.

Sentence (truncated)	中国镍、钴金属依赖进口的局面得到缓解 China's dependence on imports of nickel and cobalt has been alleviated.
Gold seg	中国, 镍、钴, 金属, 依赖, 进口, 缓解 China, nickel and cobalt, metal, dependence, imports, alleviated
Ground Truth	B E$_{(GPE)}$ O O O O O O O O O O O O O O O O 中国$_{(GPE)}$ 镍、钴金属依赖进口的局面得到缓解
FLAT+BERT	B M E$_{(GPE)}$ O O O O O O O O O O O O O O O 中国镍$_{(GPE)}$、钴金属依赖进口的局面得到缓解
BERT+ LSTM+CRF	B M E$_{(GPE)}$ O O O O O O O O O O O O O O O 中国镍$_{(GPE)}$、钴金属依赖进口的局面得到缓解
MPM-CNER (ours)	B E$_{(GPE)}$ O O O O O O O O O O O O O O O O 中国$_{(GPE)}$ 镍、钴金属依赖进口的局面得到缓解 China (GPE)'s dependence on imports of nickel and cobalt has been alleviated.

4 RELATED WORK

Recent advances in deep neural network models have greatly motivated the development of the Chinese NER. Zhang and Yang [41] proposed a lattice-structured LSTM model to encode the Chinese characters, which can use both the word and character-level information for Chinese NER. To disambiguate the recognition of entity boundaries, Li et al [17] introduced the lexicon information and simplified the complex lattice structure into a flat structure consisting of character spans. Ma et al [25] encoded the lexicon information into the character representations by incorporating all the lexicon-matched words for each input Chinese character.

However, most studies focused on the contextual semantics of input Chinese sentences but ignored the phonetic and glyphic information. Recently, some researchers have turned attention to encode the glyphs of Chinese characters as the graphic feature for Chinese NER. Meng et al [36] proposed a Tianzige-CNN to capture the graphic features of the Traditional Chinese that contain more pictographic information. Song and Sehanobish [29] also proposed a CNN-based model to incorporate the semantic and glyph information for Chinese NER. Wang et al [34] further verified that the phonetic radicals of Chinese characters can help in Chinese NER.

To absorb and fuse the multi-modal features, the multi-modal pre-training model was firstly adopted in the Vision-and-Language tasks to obtain better representations for downstream applications [3, 12, 30]. Lu et al [22] acquired the joint visual and textual embedding through a multi-modal model based on the extended BERT architecture. Chen et al [2] aligned the text and image regions during the pre-training process to learn the interaction between the two modalities. The Transformer-based pre-training model has become a paradigm for the multi-modalities in the Chinese language.

5 CONCLUSION

In this work, a novel multi-modal pre-training model for Chinese NER, with the cross-modality attention mechanism, was proposed to fuse the Chinese semantics, glyphs, and phonetics, for further improve the performance of Chinese NER. Experimental results verified that our method outperforms previous SOTA baselines and proved the effectiveness of multi-modal representations, which sheds light on exploiting the linguistic knowledge for Chinese NER.

ACKNOWLEDGMENTS

This work is supported by the National Key R&D Program of China (2019YFC1711000), the National Natural Science Foundation of China (NO. U1811461, 61572250), the Jiangsu Province Science & Technology Research Grant (BE2017155), and the Collaborative Innovation Center of Novel Software Technology and Industrialization, Jiangsu, China.

REFERENCES

[1] Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial Transfer Learning for Chinese Named Entity Recognition with Self-Attention Mechanism. In *EMNLP*.

[2] Yen-Chun Chen, Linjie Li, Licheng Yu, A. E. Kholy, Faisal Ahmed, Zhe Gan, Y. Cheng, and Jingjing Liu. 2020. UNITER: UNiversal Image-TExt Representation Learning. In *ECCV*.

[3] Jaemin Cho, Jiasen Lu, Dustin Schwenk, Hannaneh Hajishirzi, and Aniruddha Kembhavi. 2020. X-LXMERT: Paint, Caption and Answer Questions with Multi-Modal Transformers. In *EMNLP*.

[4] Harm de Vries, Florian Strub, Jérémie Mary, H. Larochelle, O. Pietquin, and Aaron C. Courville. 2017. Modulating early visual processing by language. In *NIPS*.

[5] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*.

[6] Laura Dietz. 2019. ENT Rank: Retrieving Entities for Topical Information Needs through Entity-Neighbor-Text Relations. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (2019). https://doi.org/10.1145/3331184.3331257

[7] Ruixue Ding, Pengjun Xie, Xiaoyan Zhang, Wei Lu, Linlin Li, and Luo Si. 2019. A Neural Multi-digraph Model for Chinese NER with Gazetteers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1462–1467. https://doi.org/10.18653/v1/P19-1141

[8] A. Graves and J. Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks : the official journal of the International Neural Network Society* 18 5-6 (2005), 602–10.

[9] Tao Gui, Ruotian Ma, Qi Zhang, Lujun Zhao, Yugang Jiang, and Xuanjing Huang. 2019. CNN-Based Chinese NER with Lexicon Rethinking. In *IJCAI*.

[10] Bowen Hao, Jing Zhang, H. Yin, Cuiping Li, and Hong Chen. 2021. Pre-Training Graph Neural Networks for Cold-Start Users and Items Representation. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (2021). https://doi.org/10.1145/3437963.3441738

[11] S. Hochreiter and J. Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9 (1997), 1735–1780.

[12] Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision. In *ICML*.

[13] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[14] J. Lafferty, A. McCallum, and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*.

[15] Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and M. Zhou. 2020. Unicoder-VL: A Universal Encoder for Vision and Language by Cross-modal Pre-training. In *AAAI*.

[16] Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A Unified MRC Framework for Named Entity Recognition. In *ACL*.

[17] Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. FLAT: Chinese NER Using Flat-Lattice Transformer. In *ACL*.

[18] Yan Li, Tingjian Ge, and Cindy Chen. 2020. Online Indices for Predictive Top-k Entity and Aggregate Queries on Knowledge Graphs. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. 1057–1068.

[19] Wei Liu, Xiyan Fu, Yue Zhang, and Wenming Xiao. 2021. Lexicon Enhanced Chinese Sequence Labeling Using BERT Adapter. In *ACL/IJCNLP*.

[20] Wei Liu, Tongge Xu, QingHua Xu, Jiayu Song, and Yueran Zu. 2019. An Encoding Strategy Based Word-Character LSTM for Chinese NER. In *NAACL*.

[21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030* (2021).

[22] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *NeurIPS*.

[23] Xue Mengge, Bowen Yu, Tingwen Liu, Yue Zhang, Erli Meng, and Bin Wang. 2020. Porous Lattice Transformer Encoder for Chinese NER. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.340

[24] Haiyun Peng, Yukun Ma, Soujanya Poria, Yang Li, and E. Cambria. 2021. Phonetic-enriched Text Representation for Chinese Sentiment Analysis with Reinforcement Learning. *Inf. Fusion* 70 (2021), 88–99.

[25] Minlong Peng, Ruotian Ma, Qi Zhang, and Xuanjing Huang. 2020. Simplify the Usage of Lexicon in Chinese NER. In *ACL*.

[26] Nanyun Peng and Mark Dredze. 2015. Named Entity Recognition for Chinese Social Media with Jointly Trained Embeddings. In *EMNLP*. https://doi.org/10.18653/v1/d15-1064

[27] Jonathan Shen, Ruoming Pang, Ron J. Weiss, M. Schuster, Navdeep Jaitly, Zongheng Yang, Z. Chen, Yu Zhang, Yuxuan Wang, R. Skerry-Ryan, R. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. 2018. Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4779–4783.

[28] Ying Shen, Desi Wen, Yaliang Li, Nan Du, Haitao Zheng, and Min Yang. 2019. Path-based Attribute-aware Representation Learning for Relation Prediction. In *SDM*.

[29] C. Song and Arijit Sehanobish. 2020. Using Chinese Glyphs for Named Entity Recognition (Student Abstract). In *AAAI*.

[30] Chen Sun, Austin Myers, Carl Vondrick, K. Murphy, and C. Schmid. 2019. VideoBERT: A Joint Model for Video and Language Representation Learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 7463–7472.

[31] Zijun Sun, Xiaoya Li, Xiaofei Sun, Yuxian Meng, Xiang Ao, Qing He, Fei Wu, and Jiwei Li. 2021. ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information. In *ACL/IJCNLP*.

[32] Hao Tan and Mohit Bansal. 2019. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 5100–5111. https://doi.org/10.18653/v1/D19-1514

[33] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. *ArXiv* abs/1706.03762.

[34] Yifei Wang, S. Ananiadou, and Junichi Tsujii. 2019. Improving clinical named entity recognition in Chinese using the graphical and phonetic feature. *BMC Medical Informatics and Decision Making* 19 (2019).

[35] Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, et al. 2011. Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium* (2011).

[36] Wei Wu, Yuxian Meng, F. Wang, Qinghong Han, Muyu Li, Xiaoya Li, J. Mei, Ping Nie, Xiaofei Sun, and Jiwei Li. 2019. Glyce: Glyph-vectors for Chinese Character Representations. In *NeurIPS*.

[37] Wei Ye, B. Li, Rui Xie, Zhonghao Sheng, Long Chen, and Shikun Zhang. 2019. Exploiting Entity BIO Tag Embeddings and Multi-task Learning for Relation Extraction with Imbalanced Data. In *ACL*.

[38] Jianfei Yu, Jing Jiang, Li Yang, and Rui Xia. 2020. Improving Multimodal Named Entity Recognition via Entity Span Detection with Unified Multimodal Transformer. In *ACL*.

[39] Jianshu Zhang, Jun Du, and Lirong Dai. 2020. Radical analysis network for learning hierarchies of Chinese characters. *Pattern Recognit.* 103 (2020), 107305.

[40] Suxiang Zhang, Ying Qin, Juan Wen, and Xiaojie Wang. 2006. Word Segmentation and Named Entity Recognition for SIGHAN Bakeoff3. In *SIGHAN@COLING/ACL*.

[41] Yue Zhang and Jie Yang. 2018. Chinese NER Using Lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 1554–1564. https://doi.org/10.18653/v1/P18-1144

[42] Zhuosheng Zhang, Yafang Huang, and Hai Zhao. 2019. Open Vocabulary Learning for Neural Chinese Pinyin IME. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1584–1594. https://doi.org/10.18653/v1/P19-1154

[43] Z. Zhang, Zhifei Li, Hai Liu, and Neal Xiong. 2020. Multi-scale Dynamic Convolutional Network for Knowledge Graph Embedding. *IEEE Annals of the History of Computing* (2020), 1–1.