Contextual Characters with Segmentation Representation for Named Entity Recognition in Chinese

Baptiste Blouin, Pierre Magistry

Contextual Characters with Segmentation Representation for Named Entity Recognition in Chinese

BLOUIN Baptiste
Aix-Marseille University, IrAsia
Aix-Marseille University, LIS
ENP-China
baptiste.blouin@lis-lab.fr

MAGISTRY Pierre
Aix-Marseille University, IrAsia
ENP-China
pierre@magistry.fr

Abstract

Named Entity Recognition (NER) is a typical sequence labeling task. It remains challenging for Chinese, partly because of the lack of clear typographic word boundaries. Decisions have to be made regarding the choice of basic units which constitute the sequence to be labeled, and their vectorized representation. Recent approaches have shown that character-based models lack the information about larger units (words) which is useful for NER, while word-based models may suffer from the propagation of word segmentation errors and a higher rate of Out-of-Vocabulary (OOV) tokens. In this paper, we propose a new representation of sinograms (Chinese characters) enriched with word boundary information, for which different types of embeddings can be built. Experiments show that our solution outperforms other state-of-the-art models. We also took great care to propose a fully retrainable pipeline, which is available at https://github.com/enp-china/CCSR-NER . It does not rely on pretrained models and can be trained in few days on common hardware.

1 Introduction

The present work explores the task of Named Entity Recognition (NER) in Mandarin Chinese, specifically for cases when relying on large pre-trained models is not an option. This can occur when one has to process domain specific data, or in our case[1], historical texts where language is quite different from the language of the corpora used to pretrain publicly available models, especially words and characters embeddings. The models we propose can be trained in a reasonable time (days) from a relatively small amount of raw data (few hundred millions of characters) on affordable hardware (such as a single GTX 1080 ti).

[1]ENP China, https://www.enpchina.eu/ (ERC No 788476)

Chinese script does not provide a clear and frequent typographic marker for word boundaries. As a result, when addressing the case of Chinese(s) language(s) in NER, we have to face the issue of word segmentation. Recent models proposed in the literature can be divided into character-based, word-based or hybrid models, but every work had to take a stance regarding Chinese Word Segmentation (CWS). The importance and methods for CWS have a long history in Chinese NLP, a recent work Li et al. (2019) makes the strong claim that the neural era of NLP is turning CWS into an irrelevant or even harmful step in a pipeline. However Li et al. (2019) did not provide experimental results on the NER task and our own experiments presented in this paper tend to show that CWS can be either harmful or beneficial, depending on how much care is given to consistency in segmentation and to the way word embeddings are built and used. Our main findings are that off-the-shelf embeddings for Mandarin Chinese must be used carefully, but it is possible to improve on the state-of-the-art by retraining everything from raw and labeled corpora, as we achieve 77.27 (+2.84) of f-score on OntoNotes 4 (Hovy et al., 2006) and 80.64 (+1.04) on OntoNotes 5 with a model simpler than previous state-of-the-art which requires dependency parsing.

The second focus of our study is a comparison between supervised and unsupervised CWS. When targeting a specific downstream NLP task, we ran experiments to decide whether we should follow a specific segmentation guideline by the mean of supervised machine learning, or if consistency brought by an unsupervised system is enough to improve on the downstream (here NER) task. This question is crucial for us to face more ancient texts, for which training data for CWS may not be available. We show that using CWS for the task of named entity recognition allows to provide useful information compared

to using only characters.

In summary, the contributions of this paper are as follows:

- We propose a novel method to combine CWS information and a character-level representation which can be used by a BiLSTM-CRF (Lample et al., 2016) model to improve on Chinese NER task.
- In an attempt to explain this improvement, we study the impact of our new representation on the OOV issue compared to other possible representations.
- We investigate two different strategies of supervised and unsupervised CWS, to assess for the need of manually segmented training corpus.
- The experimental results demonstrate that our proposed method significantly outperforms the current state-of-the-art performance on five different Chinese NER datasets. Our proposed solution does not rely on any pre-trained models, and can be fully trained from corpora of relatively small size on affordable hardware.

2 Related works

Our work relates to existing methods on multiple tasks, including NER, segmentation and embeddings.

2.1 Named Entity Recognition

Our model architecture is similar to that proposed by Huang et al. (2015), which is a bidirectional recurrent neural network (BiLSTMs) with a subsequent conditional random field (CRF) decoding layer. For this kind of architecture we have to choose a level of tokenization for the input. It can result in word-based models, character-based models and hybrid models. A word-based BiLSTM-CRF model applied to Chinese NER will suffer from segmentation errors. Zhang and Yang (2018) and Liu et al. (2019) showed that using a hybrid model to integrate words in character sequence leads to better results for character-based Chinese NER. The main difference between those models is that Zhang and Yang (2018) uses a DAG-structured LSTM to put every potential words that match a lexicon into their model, this requires them to process sentences one by one, whereas Liu et al. (2019) add word information into the input vector. This second approach selects a single segmentation and choose one word for each character without ambiguity.

Another approach to integrate the word segmentation information to the model was proposed by Cao et al. (2018) which involves using multitask on Chinese segmentation to transfer this information to the NER task.

Jie and Lu (2019) propose a more complex approach which integrates dependency parses to the LSTM and relies on pre-trained ELMo contextual embeddings. They obtain promising results on the OntoNotes 5 corpus, but they do not discuss the issue of word segmentation (for which they use the gold segmentation).

2.2 Word Segmentation

Word-level information can be introduced into a NER system in various ways, as a first step of processing or to build an external resource such as a word embeddings lexicon. In any case, it relies on a Chinese Word Segmentation (CWS) system and training corpus in the supervised case. When using pre-trained word embeddings, one implicitly relies on the CWS system which has been used to prepare the embeddings. In our case we conduct two kinds of experiments, the first one is based on supervised CWS for which we use *zpar* (Zhang and Clark, 2007) trained on the Chinese Treebank[1]. Since training data for word segmentation is not available for all domains, languages (to adapt to other sinitic languages, such as Cantonese) or more ancient documents, and can be time consuming or costly to obtain, we also run experiments based on an unsupervised CWS system using *eleve* (Magistry and Sagot, 2012) which requires only an unannotated corpus. We use texts from the Chinese Wikipedia to train the segmenter, which we sampled from the corpus prepared by Majliš and Žabokrtský (2012) down to a size we think consistent to what will be available for future adaptations of our system.

2.3 Embeddings

Vectorized word representations (Turian et al., 2010; Mikolov et al., 2013), especially known as word embeddings, are a key element for multiple NLP

[1] https://catalog.ldc.upenn.edu/LDC2013T21

tasks including NER (Collobert et al., 2011). Today there are three distinct embedding types. Classical word embedding (Pennington et al., 2014; Mikolov et al., 2013) , character-level features (Ma and Hovy, 2016; Zhang and Yang, 2018) and contextualized word embeddings (Peters et al., 2017; Zhang and Yang, 2018). Contextualized word embeddings as been shown to be effective for improving many natural language processing tasks including NER. In our work we use FastText (Bojanowski et al., 2016a) to generate our non-contextual embeddings and Flair Akbik et al. (2018) for the contextual ones. We decided not to use BERT (Devlin et al., 2018) because in our situation we will have to train new embeddings on multiple historical subcorpora of a limited size, which makes BERT either unusable or not affordable. It remains worth noting that we outperform the systems tested in (Jie and Lu, 2019) which rely on ELMo (Peters et al., 2018) and for which the authors report it obtained performances similar to BERT in preliminary experiments.

3 Datasets

The larger project for which we design our models introduces constraints in terms of corpus size and retrainability. We limit ourselves to a reasonable amount of data. Nevertheless, for the experiments presented in this paper, we rely on standard datasets of Modern Chinese, widely used in the literature to be able to provide a comprehensive evaluation.

We limit our raw data to a random sample of 324 millions tokens (243 millions sinograms) taken from the Wikipedia in Mandarin Chinese. We make this sample available for the sake of reproducibility.

For word segmentation, we finally used the Chinese Treebank (CTB) [1] and compare it to an unsupervised word segmentation.[2]

For Named Entities, we use the OntoNotes4 Corpus (Hovy et al., 2006) and follow the de facto standard split and entity types selection from Che et al. (2013). We also evaluate our system against the popular MSRA (Levow, 2006) Weibo NER (Peng and Dredze, 2015) and corpus of resume in Chinese (Zhang and Yang, 2018). Those four datasets represent three different domains, OntoNotes and MSRA datasets are in the news domain, the Chinese resume dataset contains resumes of senior executives from listed companies in the Chinese stock market and the Weibo NER dataset is drawn from the social media website Sina Weibo. Another difference between those datasets is that MSRA , Weibo and Chinese resume did not provide word segmentation for all the sections, unlike OntoNotes4 which has a gold-standard segmentation for the training, development and test sections. We also provide results on OntoNotes5 (Weischedel et al., 2013) to compare our system with Jie and Lu (2019). We summarize the datasets in Table 1.

Dataset	Type	Train	Test	Dev
OntoNotes4 18 classes	Sent	15.7k	4.3k	4.3k
	Char	491.9k	208.1k	200.5k
	Entities	13.4k	7.7k	6.95k
OntoNotes5 18 classes	Sent	38.3k	4.3k	6.3k
	Char	1212k	145k	175k
	Entities	64.1k	7.6k	9.2k
Weibo 4 classes	Sent	1.4k	0.27k	0.27k
	Char	73.8k	14.8k	14.5k
	Entities	1.89k	0.42k	0.39k
Resume 8 classes	Sent	3.8k	0.48k	0.46k
	Char	124.1k	15.1k	13.9k
	Entities	1.34k	0.15k	0.16k
MSRA 3 classes	Sent	46.4k	4.4k	-
	Char	2169.9k	172.6k	-
	Entities	74.8k	6.2k	-

Table 1: Statistics of the datasets

4 Methods

4.1 Contextual Character Embeddings

Contextual word embeddings have shown to improve state-of-the-art on several NLP tasks. One of our contribution is to propose two new kinds of contextual embeddings at the character level which can take into account word boundary information.

Referring to Akbik et al. (2018) paper which introduces a word-level embeddings based on a character-level language model, we introduce a sinogram embedding using their character language model (LM). Where the LM allows the text to be treated as a sequence of characters passed to an LSTM which at each point in the sequence is trained

[2] we also tried to use the dataset from Peking University (PKU) and Microsoft Research (MSR) provided for the CWS Bakeoff 2 (http://sighan.cs.uchicago.edu/bakeoff2005/) but it did not make any noticeable differences.

to predict the next character. In our system, we train the LM to produce characters with segmentation information. Given a sequence of characters ($C_0, C_1, ..., C_N$) we learn $P(C_i|C_0, ..., C_{i-1})$, an estimate of the predictive distribution over the next character given past characters. We utilize the hidden states of a forward-backward recurrent neural network to create contextualized character embeddings. The final contextual character representation is given by :

$$C_i^{LM} = \begin{bmatrix} C_i^f \\ C_{T-i}^b \end{bmatrix}$$

Where C_i^f denote the hidden state at position i of the forward LM and C_{T-i}^b denote the hidden state at position $T - i$ of the backward LM.

4.2 Contextual Character with segmentation information Embeddings

In this work, we investigate the different ways to inject the CWS information into a NER pipeline. Several approaches propose to directly use the word-tokens as segmented by a CWS system, they showed that discrepancies between the output of the CWS and the NE annotation can be harmful for NER. Out-of-Vocabulary (OOV) tokens is another common issue for NER. In order to tackle those issues, we designed a new kind of sinogram representation which contains the information of the chosen word segmentation at the character level. We decide to use the BIES format to represent the CWS (as introduced in Xue and Shen (2003), originally as an intermediary step for CWS) and we train a language model to produce embeddings of those character with BIES tag. As we use a BI-LSTM to process the NER task and as we stay at a character level, our new representation allows us to reconstruct the entire word according to the BIES tag. But in the case of a mismatching segmentation between NE and word, the model can still learn to use this wrong segmentation as the right delimiter of an entity.

4.3 Model Description

We use the Flair framework (Akbik et al., 2019) to create our model (Figure 2.1). The main difference with other existing NER models is that we use stacked embeddings to represent our input. With this kind of architecture we can combine our different kinds of embeddings. Character, word information and bichar embeddings are concatenated to represent each character. The final character representation is given by

$$c_i = \begin{bmatrix} r_i^{char} \\ r_i^{bichar} \\ r_i^{word} \end{bmatrix}$$

The fact that we use character as neural units allows us to give word information associated to a character. In our case, the word information is given by the contextual character with segmentation embeddings. We denote a Chinese sentence as s = $\{c_1, c_2, ..., c_n\}$.

We use an extra linear layer between the input layer and the LSTM's to make the stacked representation trainable. Figure 1 shows the structure of our model. The blue part of the model shows how we use the embeddings. The symbol $\bigoplus$ indicates the possibility to concatenate different kinds of embeddings. Using this approach, we can then add other types of embeddings related to characters. The red part is a BiLSTM-CRF.

5 Experiments

We conducted several experiments to evaluate the effectiveness of our approach across different domains. In addition, we evaluate the importance of the segmentation for our representations by using supervised and non-supervised segmentation approaches. We also investigate on the usefulness of the bichar representation for Chinese Natural Language Processing. Evaluations are reported using standard metrics of precision (P), recall (R) and F1-score (F).

5.1 Experimental Settings

We used the datasets presented in the section 3, including the OntoNotes gold segmentation to evaluate the distance between our supervised/unsupervised segmentations and whether this distance makes a difference to our overall process.

Embeddings. We used FastText (Bojanowski et al., 2016b) to pretrain characters and bi-characters embeddings on a subset of 7 millions sentences from Chinese Wikipedia dump. for both of these representations we used a context of bi-character.

Hyper-parameter. Table 2 shows the values of hyper-parameters for our models, which were fixed

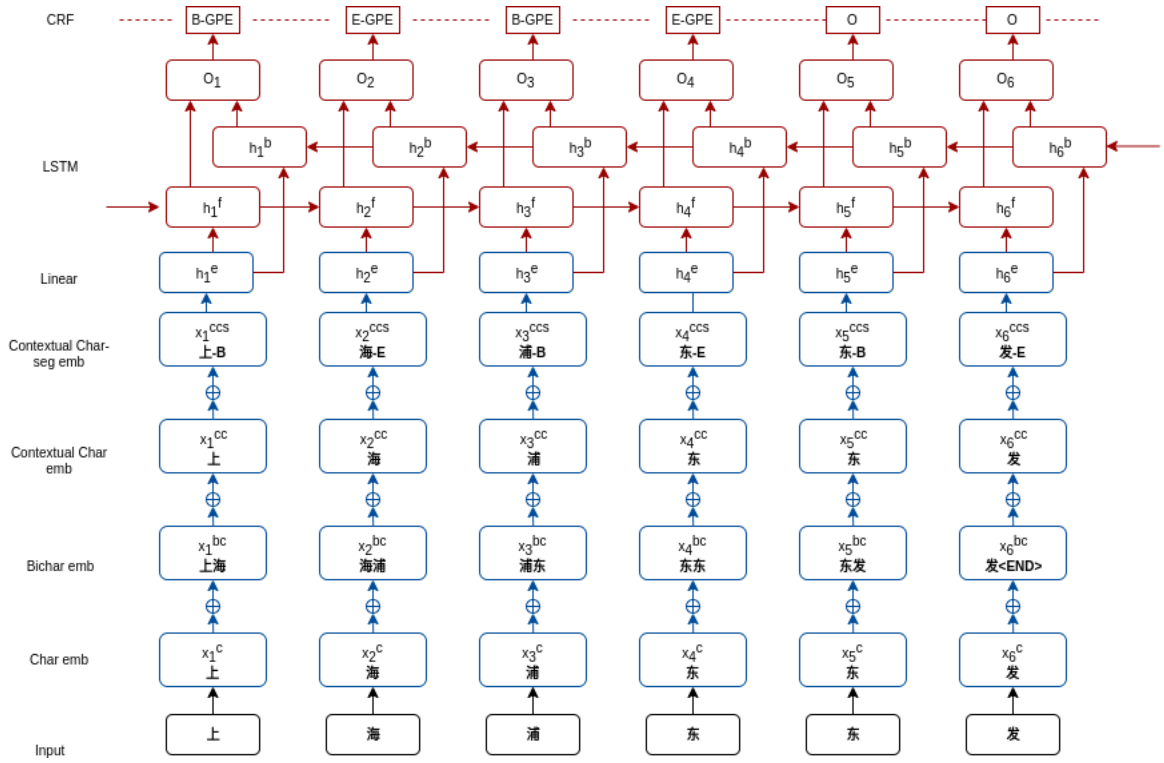

Figure 1: Architecture of the model and representation of our embeddings

Parameter	value	Parameter	value
Char emb size	50	Bichar emb size	50
LSTM hidden	256	LSTM layer	1
Learning rate	0.1	Anneal factor	0.5
Emb dropout	0.05	batch size	16

Table 2: Hyper-parameter values.

without specific grid search adjustments for each individual dataset. Stochastic gradient descent (SGD) is used for optimization, with an initial learning rate of 0.1 and we divide its value by two if the f-score does not increase on the development corpus during 5 epochs. In that case, we reload the previous best model before dividing the learning rate.

Configurations. In order to evaluate the importance of the different representations, we have set up 8 configurations of embeddings.

- **Char** For this configuration we only use character embeddings. (size : 50)
- **Bichar** For this configuration we only use bi-character embeddings. (size : 50)
- **Bichar + Char** For this configuration we concatenate bi-character and character embeddings. (size : 50 + 50)
- **Char ctx** For this configuration we only use contextual character embeddings. (size : 1024)
- **Char-seg unsup** For this configuration we only use contextual character with segmentation information embeddings where the segmentation comes from the unsupervised segmenter. (size : 1024)
- **Bichar + Char-seg unsup** For this configuration we only concatenate bi-character embeddings to the previous configuration. (size : 50 + 1024)
- **Char-seg ctb** for this configuration we only use contextual character with segmentation information embeddings where the segmentation comes from the supervised segmenter trained on the Chinese Treebank. (size : 1024)
- **Bichar + Char-seg ctb** or this configuration we only concatenate bi-character embeddings to the previous configuration. (size : 50 + 1024)

Input	Models	P	R	F
Gold seg	Wang et al. (2013)	76.43	72.32	74.32
	Che et al. (2013)	77.71	72.51	**75.02**
	Yang et al. (2017)	65.59	71.84	68.57
No seg	Zhang and Yang (2018)	76.35	71.56	73.88
	Char baseline	70.08	60.53	64.95
	Liu et al. (2019)	76.09	72.85	74.43
	Char	67.31	64.33	65.79
	Bichar	72.25	72.18	72.21
	Bichar + Char	74.11	72.75	73.42
	Char ctx	76.79	75.66	76.22
	Char-seg unsup *CSU*	77.54	75.91	76.72
	Bichar + *CSU*	76.3	76.77	76.53
	Char-seg ctb *CSC*	77.81	76.21	77
	Bichar + *CSC*	77.67	76.89	**77.27**[3]

Table 3: NER results for named entities on the OntoNotes 4 dataset. There are three blocks. The first two blocks contain the previous state-of-the-art models where "Gold seg" means that they used the reference segmentation proposed by the dataset and "No seg" means that they used other approaches that do not rely on reference segmentation. The last block lists the performance of our proposed model.

5.2 Experimental results

OntoNotes. Table 3 shows the experimental results on OntoNotes 4 dataset. The first column (Input) shows the representations of input sentence that was used. "Gold seg" means that they used the segmentation provided by the corpus to represent the word in the sentence, "No seg" means that we used only the character as input and other approaches that do not benefit from the reference segmentation to provide information about the word level.

The first part of table 3 are the results of Wang et al. (2013); Che et al. (2013); Yang et al. (2017) . These three approaches rely on gold segmentation at the word level, with character embeddings. Che et al. (2013) achieve good performance with 75.02 F-score. Here we exceed this score without using the gold segmentation.

The second part shows the performances of more recent approaches (Zhang and Yang, 2018; Liu et al., 2019) and a character baseline which is the original character-based BILSTM-CRF model. Zhang and Yang (2018) proposes a lattice LSTM to exploit word information in character sequence and Liu et al. (2019) use a new word-character LSTM model to add word information on the first or on the last character of each word. These two approaches show a significant improvement compared to the character baseline, which illustrates the importance of the word information in character sequence.

The last part of the table 3 shows the results of our configurations. The first three rows show results where we only used the character information. Through these results we show that bichar representations are very efficient for Chinese. This may be explained by the fact that bichars have a length closer to the average word length and provide more contextual information than single characters. The last four rows show the results of using our contextual char-seg representations. Those configurations achieve very good results, improving the state of the art, beating both models that do not use gold segmentation and even those that do. Firstly, these results show that the information about the boundaries of a word is useful. Secondly, on this corpus, we can see that there is only a slight difference between using supervised and unsupervised segmentation. Which is very encouraging to address situations where we do not have adequate CWS training data.

Weibo NER. Table 4 shows the experimental results on Weibo NER dataset. This dataset proposes two kinds of annotations, named entities and nominal entities. For our experiments we only evaluated the combination of these two annotations. Compared to the other corpus, this one offers few annotated data, that is why different approaches have been proposed. Peng and Dredze (2015, 2016); Cao et al. (2018) use multitask learning and He and Sun (2017) use semi-supervised learning. As a result of these approaches, they use cross-domain or semi-supervised additional data. In contrast, Zhang and Yang (2018); Liu et al. (2019) and our model do not need any additional data.

These results exhibit similar patterns as those on OntoNotes. However in this case the unsupervised CWS can even lead to higher scores. This may be the result of Weibo Corpus being drawn from social media. A CWS system trained on the CTB is better suited for the news domain and less reliable in the Weibo case.

[3]This result is the average of 20 runs. The results of these runs have a variance of 4.10^{-2}

Models	P	R	F
Peng and Dredze (2015)	-	-	56.05
Peng and Dredze (2016)	-	-	58.99
He and Sun (2017)	-	-	58.23
He and Sun (2017)	-	-	54.82
Cao et al. (2018)	-	-	58.70
Zhang and Yang (2018)	-	-	58.79
Liu et al. (2019)	-	-	59.84
char baseline	-	-	52.88
Char	72.14	34.69	46.85
Bichar	72.63	33.01	45.39
Bichar Char	69.73	49.04	57.58
Char ctx	66.67	52.63	58.82
Char-seg unsup	66.48	55.98	60.78
Bichar + Char-seg unsup	70.37	59.09	**64.24**
Char-seg ctb	71.25	55.74	62.55
Bichar + Char-seg ctb	67.24	56.46	61.38

Table 4: Weibo NER results

Models	P	R	F
Zhang and Yang (2018)	94.81	94.11	94.46
Liu et al. (2019)	95.27	95.15	95.21
char baseline	93.26	93.44	93.35
Char	92.76	94.36	93.55
Bichar	93.64	94.79	94.21
Bichar Char	93.93	94.97	94.45
Char ctx	94.39	95.03	94.71
Char-seg unsup	94.77	95.58	95.17
Bichar + Char-seg unsup	94.56	94.91	94.73
Char-seg ctb	94.84	94.66	94.75
Bichar + Char-seg ctb	95.07	95.83	**95.45**

Table 5: Chinese resume results

Models	P	R	F
Zhang et al. (2006)	92.20	90.18	91.18
Zhou et al. (2013)	91.86	88.75	90.28
Dong et al. (2016)	91.28	90.62	90.95
Cao et al. (2018)	91.73	89.58	90.64
Zhang and Yang (2018)	93.57	92.79	93.18
Liu et al. (2019)	94.33	93.11	93.71
char baseline	89.61	86.98	88.37
Char	84.95	84.37	84.66
Bichar	87.3	83.74	85.48
Bichar Char	90.13	89.74	89.93
Char ctx	90.6	88.58	89.58
Char-seg unsup	94.77	93.43	94.1
Bichar + Char-seg unsup	94.93	93.38	**94.15**
Char-seg ctb	93.63	91.42	92.51
Bichar + Char-seg ctb	93.73	91.78	92.74

Table 6: MSRA results

Models	P	R	F
Zhang and Yang (2018)	76.34	77.01	76.67
Jie and Lu (2019)			
BiLSTM-CRF	77.94	75.33	76.61
BiLSTM-CRF + ELMo	79.20	79.21	79.20
DGLSTM-CRF + ELMo	78.86	81.00	79.92
without Gold dep.			79.59
Bichar + Char-seg ctb	80.70	80.60	**80.65**

Table 7: Ontonotes 5 results. Jie and Lu (2019) provide detailed results on gold segmentation and parsing only. An F-measure of 79.59 is obtained with non-gold dependencies, but the authors did not report experiments related to the quality of the word segmentation.

Resume Table 5 shows the experimental results on Resume dataset. These are consistent with the observations made on OntoNotes and Weibo NER. Our model achieves good results on this dataset, but unlike the other corpora, very good results were already obtained by other systems. It does not allow us to highlight our approach as much as the other corpora.

MSRA Table 6 shows the experimental results on MSRA dataset. The best results are obtained with the unsupervised segmentation.

Ontonotes 5 To complete our evaluation, we run our best model from the Ontonotes 4 experiment on Ontonotes 5 to provide comparison with Jie and Lu (2019). Results are shown Table 7. Note that the comparison is somewhat unfair as Jie and Lu (2019) rely on gold segmentation. Nevertheless, our system obtains the highest results, without the need for a dependency parser.

The embeddings we propose achieve state-of-the-art results on a diversity domains such as news, social media, and Chinese resume.

5.3 Out Of Vocabulary analysis

When using a model with word-level features, one of the most common problems comes from unknown words. Our approach which injects segmentation information at the characters level allows to rebuild the words from characters and leads to fewer unknowns.

To do so, we used two types of segmentation, word level and char-seg level, in a supervised and unsupervised way to segment our Wikipedia sample. Once our four Wikipedia samples were segmented,

embeddings	OntoNotes seg	OOV
Word ctb	supervised ctb	18.89 %
Word ctb	gold	18.96 %
Word unsup	unsupervised	32.34 %
Word unsup	gold	35.81 %
Char-seg ctb	supervised ctb	0.67 %
Char-seg ctb	gold	0.28 %
Char-seg unsup	unsupervised	0.95 %
Char-seg unsup	gold	1.78 %

Table 8: OOV statistics on OntoNotes 4 with supervised and unsupervised segmentation.

we trained four different FastText to obtain 4 lexicons for each of them. To evaluate the OOV rate on OntoNotes, we segmented it in three different ways in order to compare for each case the presence or not of words in the lexicons generated by our embeddings. We segmented OntoNotes in a supervised and unsupervised way with the same two models we used to segment Wikipedia and in a last step we left the "gold" segmentation in words proposed by OntoNotes. Results of this experiments are shown in table 8.

For the embeddings column, we have two levels of segmentation, in word and char-seg, and two levels of supervision, "ctb" for the supervised part trained on the Chinese TreeBank and "unsup" for the unsupervised part. The OntoNotes seg column represents the three types of segmentation used to segregate OntoNotes into words. Because OntoNotes is segmented into words and because our lexicon for our char-seg embeddings contains only characters with segmentation information, for a given word coming from OntoNotes, we try to reconstruct the char-seg sequence constituting this word from our embedding lexicon. For example, for the word 越南 we are looking for char-seg 越-B and 南-E in our embedding lexicon. If a char-seg is missing, then the whole word is missing too.

The results show our representations greatly decrease the unknown word rates. it allows us to have a representation for most of the words. Moreover, unlike traditional word representations, we do not have fixed representations of our words, which makes it easier to have representations for new words, but which can then call into question the quality of our representations.

6 Discussions

Annotation ambiguity. The named entity recognition task combines a step of segmentation with one of classification. We feel the need to question some cases of ambiguity from the data. By using the guideline from OntoNotes we annotated in-house data and we found it difficult in some cases to choose between Geopolitical Entity (GPE) and Location (LOC). This case of ambiguity has a direct impact on our predictions. we noted that more than $\frac{1}{3}$ of LOC that has been detected is annotated as a GPE, which is consistent with the difficulties encountered in our annotation experiment.

Another issue arises from the conversion of the OntoNotes 4 corpus from 18 classes to 4. Most notably the entity types NORP (Nationality, Other, Religion, Political) and FAC (Facility). These classes are discarded in the 4-classes version, but are typical cases of nested entities containing a GPE, LOC or ORG, which is also discarded in the process, creating erroneous annotation.

Entity segmentation against word segmentation. Our results show that although staying at the character level allows us to tackle the OOV issue, the information brought by CWS is still what enables us to reach the highest scores. In the cases when the CTB segmentation guidelines are consistent with the NER corpus, supervised segmentation performs better. However NER with unsupervised segmentation is close in these cases and can perform better in other cases. So our answer to Li et al. (2019) could be that Word Segmentation is actually necessary, but unsupervised CWS may be enough.

7 Conclusion and future works

In this paper, we propose new sinogram embeddings which includes word information at the character level for Chinese NER. Our proposed approach shows that adding CWS label to a character allows to give word level information while reducing considerably the number of OOV compared to a word sequence. Our experiments on multiple datasets, in different domains, show that our system outperforms previous state-of-the-art approaches. This paves the road to NER in more challenging situations such as historical documents or less-resourced situations.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. https://www.aclweb.org/anthology/N19-4010 FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. https://www.aclweb.org/anthology/C18-1139 Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016a. http://arxiv.org/abs/1607.04606 Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016b. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. https://www.aclweb.org/anthology/D18-1017 Adversarial transfer learning for Chinese named entity recognition with self-attention mechanism. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 182–192, Brussels, Belgium. Association for Computational Linguistics.

Wanxiang Che, Mengqiu Wang, Christopher D. Manning, and Ting Liu. 2013. https://www.aclweb.org/anthology/N13-1006 Named entity recognition with bilingual constraints. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 52–62, Atlanta, Georgia. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. http://dl.acm.org/citation.cfm?id=2078183.2078186 Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 999888:2493–2537.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. http://arxiv.org/abs/1810.04805 BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Chuanhai Dong, Jiajun Zhang, Chengqing Zong, Masanori Hattori, and Hui Di. 2016. https://doi.org/10.1007/978-3-319-50496-4_20 Character-based lstm-crf with radical-level features for chinese named entity recognition. volume 10102, pages 239–250.

Hangfeng He and Xu Sun. 2017. https://www.aclweb.org/anthology/E17-2113 F-score driven max margin neural network for named entity recognition in Chinese social media. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 713–718, Valencia, Spain. Association for Computational Linguistics.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. http://dl.acm.org/citation.cfm?id=1614049.1614064 Ontonotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, NAACL-Short '06, pages 57–60, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. http://arxiv.org/abs/1508.01991 Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Zhanming Jie and Wei Lu. 2019. https://doi.org/10.18653/v1/D19-1399 Dependency-guided LSTM-CRF for named entity recognition. 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), pages 3862–3872, Hong Kong, China. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. https://doi.org/10.18653/v1/N16-1030 Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Gina-Anne Levow. 2006. https://www.aclweb.org/anthology/W06-0115 The third international Chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 108–117, Sydney, Australia. Association for Computational Linguistics.

Xiaoya Li, Yuxian Meng, Xiaofei Sun, Qinghong Han, Arianna Yuan, and Jiwei Li. 2019. https://doi.org/10.18653/v1/P19-1314 Is word segmentation necessary for deep learning of Chinese representations? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3242–3252, Florence, Italy. Association for Computational Linguistics.

Wei Liu, Tongge Xu, Qinghua Xu, Jiayu Song, and Yueran Zu. 2019. https://www.aclweb.org/anthology/N19-1247 An encoding strategy based word-character LSTM for Chinese NER. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2379–2389. Association for Computational Linguistics.

Xuezhe Ma and Eduard Hovy. 2016. http://arxiv.org/abs/1603.01354 End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *arXiv e-prints*, page arXiv:1603.01354.

Pierre Magistry and Benoît Sagot. 2012. https://www.aclweb.org/anthology/P12-2075 Unsupervized word segmentation: the case for Mandarin Chinese. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 383–387, Jeju Island, Korea. Association for Computational Linguistics.

Martin Majliš and Zdeněk Žabokrtský. 2012. http://www.lrec-conf.org/proceedings/lrec2012/pdf/267_Paper.pdf Language richness of the web. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2927–2934, Istanbul, Turkey. European Language Resources Association (ELRA).

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Nanyun Peng and Mark Dredze. 2015. https://doi.org/10.18653/v1/D15-1064 Named entity recognition for Chinese social media with jointly trained embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 548–554, Lisbon, Portugal. Association for Computational Linguistics.

Nanyun Peng and Mark Dredze. 2016. http://arxiv.org/abs/1608.02689 Multi-task multi-domain representation learning for sequence tagging. *CoRR*, abs/1608.02689.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. https://doi.org/10.18653/v1/N18-1202 Deep

contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. http://arxiv.org/abs/1705.00108 Semi-supervised sequence tagging with bidirectional language models. *CoRR*, abs/1705.00108.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. http://dl.acm.org/citation.cfm?id=1858721 Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mengqiu Wang, Wanxiang Che, and Christopher D. Manning. 2013. http://dl.acm.org/citation.cfm?id=2891460.2891588 Effective bilingual constraints for semi-supervised learning of named entity recognizers. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, AAAI'13, pages 919–925. AAAI Press.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. https://catalog.ldc.upenn.edu/LDC2013T19 Ontonotes release 5.0. *Linguistic Data Consortium*.

Nianwen Xue and Libin Shen. 2003. https://doi.org/10.3115/1119250.1119278 Chinese word segmentation as lmr tagging.

Jie Yang, Zhiyang Teng, Meishan Zhang, and Yue Zhang. 2017. http://arxiv.org/abs/1708.07279 Combining discrete and neural features for sequence labeling. *CoRR*, abs/1708.07279.

Suxiang Zhang, Ying Qin, Juan Wen, and Xiaojie Wang. 2006. https://www.aclweb.org/anthology/W06-0126 Word segmentation and named entity recognition for SIGHAN bakeoff3. In *Proceedings of the Fifth SIGHAN Workshop on Chinese Language Processing*, pages 158–161, Sydney, Australia. Association for Computational Linguistics.

Yue Zhang and Stephen Clark. 2007. https://www.aclweb.org/anthology/P07-1106 Chinese segmentation with a word-based perceptron algorithm. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 840–847, Prague, Czech Republic. Association for Computational Linguistics.

Yue Zhang and Jie Yang. 2018. https://doi.org/10.18653/v1/P18-1144 Chinese ner using lattice lstm. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1554–1564, Melbourne, Australia. Association for Computational Linguistics.

J. Zhou, W. Qu, and F. Zhang. 2013. Chinese named entity recognition via joint identification and categorization. *Chinese Journal of Electronics*, 22:225–230.