Twisted Recurrent Network for Named Entity Recognition

Zefu Lu[*]
University of Illinois, Urbana-Champaign
zefulu2@illinois.edu

Lei Li, **Wei Xu**
Baidu Research, Institute of Deep Learning
{lilei22,xuwei06}@baidu.com

Named entity recognition Recognizing entities in sentences is one basic task in natural language understanding. It is often a prerequisite step in larger problems such as question answering, conversation, voice search, etc. Given a sequence of text tokens, a named entity recognizer (NER) shall identity the chunks of tokens that belongs to predefined category of entities such as persons and organizations. NER problem is formulated as to produce a sequence of entity labels, one for every token in the sentence.

```
John      has lived in Britain for 14         years      .
B-PERSON   O    O   O  B-GPE    O  B-CARDINAL I-CARDINAL O
```

The problem seems rather easy with a prepared database of all kinds of entities. However, it is quite challenging due to two reasons: 1. entity databases are often incomplete (considering the number new organizations established every day); 2. the same phrase can refer to different entities (or none entity) depending on context, therefore lexical form is insufficient to determine. In languages like Chinese, the problem is even harder with the lack of direct surface separation of words.

In this paper, we propose twisted recurrent network (Twinet), a novel model for named entity recognition (NER). Our intuition is based on an observation that every word in a sentence could contribute in recognizing whether a chunk is a named entity or not. Such contribution could extends long beyond short-range context words, and could come from reverse order of a sentence. Our method also benefits from its holistic decoding approach – the neighboring entity labels regulate each other. Our main focus is on Chinese NER though our proposed method applies to English as well. To this end, Twinet incorporates only one additional feature for Chinese – a word separation feature. The state-of-the-art word separation algorithm can already achieve accuracy of over 95% Ma & Chen (2003). Twinet does not require any predefined entity database.

Tokenization and embedding Twinet takes two input sequences: 1. original sequence of tokens – words in English and characters in Chinese; 2. additional features – letter caption feature (Collobert et al. (2011)) for English and word separation indicator for Chinese.

Elements in the two sequences are mapped into d_1 and d_2-dimensional vectors respectively through two look-up tables. The combined vectors are then transformed through a nonlinear perceptron layer. The result is a sequence of vectors representing token-wise states.

Twisting RNNs Twinet exploits two parallel branches, each is composed of a recurrent network layer, a nonlinear perceptron layer, and reversed recurrent network layer. Note that branches are "twisted" – the order of the layers is reversed in the second branch, and all the output of those recurrent layers are collected in the end.

A recurrent neural network (RNN) takes a sequence of input vectors $x_{1..T}$, and recurrently computes hidden states (also output).

$$h_t = \sigma(\mathrm{U} \cdot x_t + \mathrm{W} \cdot h_{t-1}), \qquad t = 1..T$$

Where $\sigma(\cdot)$ is a nonlinear activation function. In our experiment, we used rectified linear units (RELU).

Maximum Entropy In order to decode the labels for each token, the twisted vectors are projected to k-dimensional output vectors. Then, the maximum entropy criterion is employed against the true labels. This is

[*] The work was performed while the author was at Baidu.

Table 1: OntoNotes 5.0 Statistics

Dataset	Training	Validation	Testing
Chinese(#character)	1933543	193330	244533
English(#word)	1644222	251043	172077

Table 2: NER performance on OnteNote 5.0 dataset.

(a) Chinese

	P	R	F1
Stanford NER	78.20	66.45	71.85
Twinet	78.69	70.54	**74.39**

(b) English

	P	R	F1
Stanford NER	84.04	80.86	82.42
Illinois NER	85.86	84.20	85.02
Twinet	86.06	86.34	**86.20**

equivalent to maximizing the conditional log-likelihood of the labels y given input sequence x.

$$l(\theta) = \sum_{i=1}^{N}\sum_{t=1}^{T}\Big(g(x^{(i)};\theta)^T \cdot y_t^{(i)} + (y_{t-1}^{(i)})^T \cdot A \cdot y_t^{(i)}\Big) - \sum_{i=1}^{N}\log\sum_{y'}\exp\sum_{t=1}^{T}\Big(g(x^{(i)};\theta)^T \cdot y'_t + (y'_{t-1})^T \cdot A \cdot y'_t\Big)$$

Where $g(\cdot;\theta)$ is the output state from the network, and A is the transition matrix, part of θ. N is the number of samples. The labels y are assumed to follow one-hot scheme.

Experimental results In our experiments, Twinet is trained and tested on OntoNotes 5.0 dataset since it includes the largest set of labeled named entities and has a total of 18 entity classes, such as PERSON, ORGANIZATION, and PRODUCT. We used the conventional BIO annotation for the chunk labels. The standard F1 score calculated from precision and recall defined in CoNLL 2003 shared task is adopted as a metric.

We borrow the word embedding layer from SENNA project Collobert et al. (2011) for our English NER model. We use the GloVe Pennington et al. (2014) toolkit to train character-based word embedding on corpus sampled from Baidu Baike[1]. The resulting look-up table is then incorporated as initial embedding in our model for Chinese NER.

We evaluated the performance of several existing models as well as our proposed Twinet. We selected two state-of-the-art methods trained on OntoNote dataset: the Stanford named entity tagger Finkel et al. (2005), which is a variant of conditional random fields (CRF), and the Illinois NER system Ratinov & Roth (2009). Wang et al. (2013) proposed joint decoding algorithm for combined word alignment and NER in English-Chinese parallel text from OntoNotes. However, it is only trained on 4 classes, therefore we did not include it in our comparison. The results are in Table **??** and **??**. Our proposed Twinet achieves significantly better results than the best existing systems.

Contribution The contribution of our paper is as follows.

- We propose twisted recurrent neural network (Twinet), a novel model for NER. It applies to other sequence labelling tasks as well;
- We identify the most effective model configuration such as (only one) additional feature and cost function;
- Our experiment on the currently largest NER datasets demonstrates that Twinet outperforms the previous state-of-the-art systems in both English and Chinese.

References

Collobert, Ronan, Weston, Jason, Bottou, Léon, Karlen, Michael, Kavukcuoglu, Koray, and Kuksa, Pavel. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November 2011. ISSN 1532-4435.

Finkel, Jenny Rose, Grenager, Trond, and Manning, Christopher. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*, 2005.

Ma, Wei-Yun and Chen, Keh-Jiann. Introduction to ckip chinese word segmentation system for the first international chinese word segmentation bakeoff. In *Proceedings of SIGHAN 2003*, pp. 168–171. ACL, 2003.

Pennington, Jeffrey, Socher, Richard, and Manning, Christopher D. Glove: Global vectors for word representation. In *EMNLP 2014*, volume 12, pp. 1532–1543, 2014.

Ratinov, Lev and Roth, Dan. Design challenges and misconceptions in named entity recognition. In *Proceedings of the CoNLL 2009*, pp. 147–155. ACL, 2009.

Wang, Mengqiu, Che, Wanxiang, and Manning, Christopher D. Joint word alignment and bilingual named entity recognition using dual decomposition. In *ACL (1)*, pp. 1073–1082, 2013.

[1] Chinese equivalent of Wikipedia, `http://baike.baidu.com`