Constructing Uyghur Named Entity Recognition System using Neural Machine Translation Tag Projection

Azmat Anwar, Xiao Li, Yating Yang, Rui Dong and **Turghun Osman**
Xinjiang Technical Institute of Physics & Chemistry, Chinese Academy of Sciences, Urumqi, China
University of Chinese Academy of Sciences, Beijing, China
Xinjiang Laboratory of Minority Speech and Language Information Processing, Urumqi, China
{azmat,xiaoli,yangyt,dongrui,turghun}@ms.xjb.ac.cn

Abstract

Although named entity recognition achieved great success by introducing the neural networks, it is challenging to apply these models to low resource languages including Uyghur while it depends on a large amount of annotated training data. Constructing a well-annotated named entity corpus manually is very time-consuming and labor-intensive. Most existing methods based on the parallel corpus combined with the word alignment tools. However, word alignment methods introduce alignment errors inevitably. In this paper, we address this problem by a named entity tag transfer method based on the common neural machine translation. The proposed method marks the entity boundaries in Chinese sentence and translates the sentences to Uyghur by neural machine translation system, hope that neural machine translation will align the source and target entity by the self-attention mechanism. The experimental results show that the Uyghur named entity recognition system trained by the constructed corpus achieve good performance on the test set, with 73.80% F1 score(3.79% improvement by baseline).

1 Introduction

Named Entity Recognition (NER) is a task of identifying named entities (NEs), especially person names (PER), location names (LOC), organization names (ORG), and classifying them into some pre-defined target entity classes (Hobbs et al., 1997). NER is essential to many natural language processing (NLP) tasks such as relation extraction (Christopoulou et al., 2019), event detection (Cakır and Virtanen, 2019), knowledge graph construction (Bosselut et al., 2019) and so on. Although the NER achieves great success by the introduction of the advanced neural networks (Collobert et al., 2011; Huang et al., 2015; Chiu and Nichols, 2016; Lample et al., 2016; Ma and Hovy, 2016; Yang et al., 2016; Peters et al., 2017; Liu et al., 2018; Peters et al., 2018), these methods are highly dependent on a large amount of annotated training data, and thus challenging to apply these models to low resource languages including Uyghur. Constructing a well-annotated NE corpus manually is very time-consuming and labor-intensive. Instead, Cross-lingual transfer is an effective solution, which addresses this challenge by transferring knowledge from a high-resource source language with abundant entity labels to a low-resource target language with few or no labels. According to the resource availability of the target language, different types of NER methods are proposed, such as bilingual parallel corpus based tag projection (Yarowsky et al., 2001; Ehrmann et al., 2011; Wang et al., 2013; Fang and Cohn, 2016; Ni et al., 2017), cross-lingual word embedding (Fang and Cohn, 2017; Wang et al., 2017; Huang et al., 2018), cross-lingual Wikification (Kim et al., 2012; Nothman et al., 2013; Tsai et al., 2016; Pan et al., 2017) or multi-task learning (Yang et al., 2016; Lin et al., 2018).

As a low resource language, Uyghur has no well-annotated corpus available for NER, but it is easy to get Uyghur-Chinese bilingual parallel corpus as Uyghur-Chinese machine translation is an important task of China Conference on Machine translation (CCMT). A common way of constructing NER corpus for the language which has a bilingual parallel corpus is using off-the-shelf NER tool in the source language to get entity annotations and transfer them to target language combing with the automatic

Proceedings of the 19th China National Conference on Computational Linguistics, pages 1007-1017, Hainan, China, October 31 - Nevermber 1, 2020.
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

word alignment. Although some researchers have also applied this method to transfer NE annotations from Chinese to Uyghur and achieved remarkable results (Maimaiti et al., 2018), these pipeline methods inevitably introduce errors from the source language, including errors from NER tools and automatic word alignment. Figure 1 illustrates an Example of NER corpus construction based on the bilingual parallel corpus and automatic word alignment.

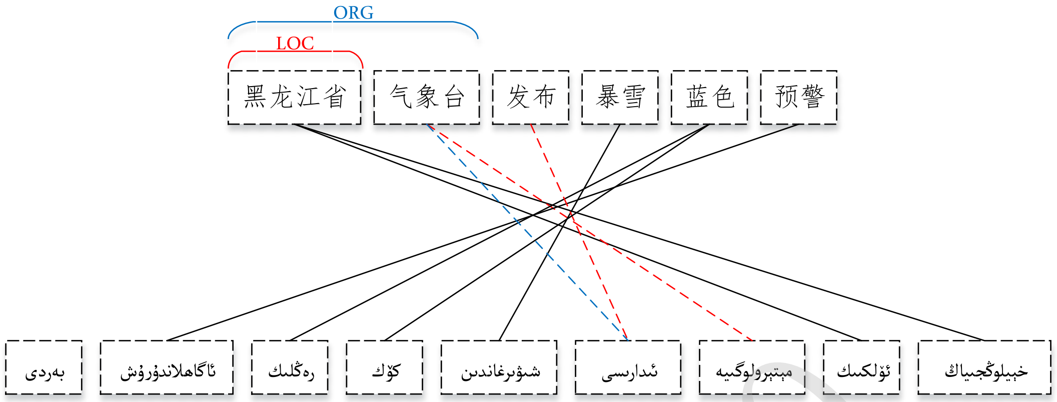

Figure 1: Example of NER corpus construction based on the bilingual parallel corpus and automatic word alignment. Errors from NER tools and automatic word alignments remarked in red color while blue indicates correct

In this paper, we address these challenges by a NE annotation transfer method based on neural machine translation (NMT). Given an Uyghur-Chinese parallel corpus, first, we train a general-purpose Chinese-Uyghur NMT system using the parallel corpus. Then, add the NE boundary information directly to the source Chinese sentence by multiple off-the-shelf NER tools. Finally, translate the Chinese sentences with entity boundary to Uyghur language using the pre-trained NMT system, we hope that NMT will align the source and target entity by the self-attention mechanism. Our method can be illustrated by the following example provided in Figure 2.

The main advantages of our method are used multi NER tools in the source language to minimize annotation errors and use general-purpose NMT without adding new tokens to indicate NE boundary in the parallel corpus, thus no need to annotate any training data manually.

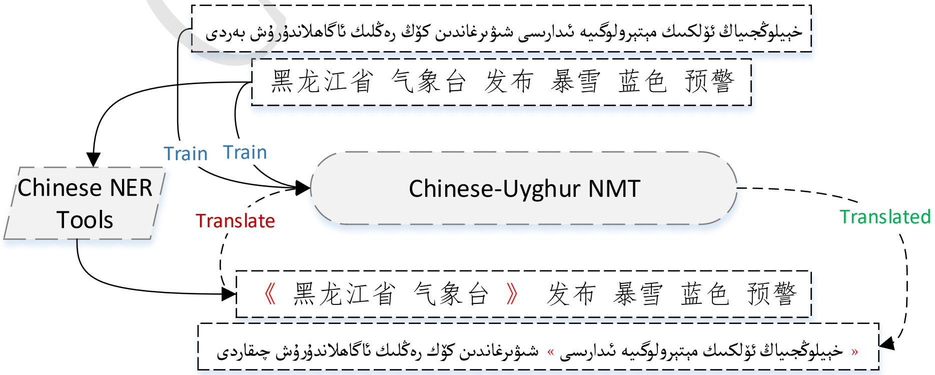
خېيلوڭجياڭ ئۆلكىلىك مېتېرولوگىيە ئىدارىسى شۋىرغاندىن كۆك رەڭلىك ئاگاھلاندۇرۇش بەردى
黑龙江省 气象台 发布 暴雪 蓝色 预警
Train
Train
Chinese NER Tools
Chinese-Uyghur NMT
Translate
Translated
《 黑龙江省 气象台 》 发布 暴雪 蓝色 预警
« خېيلوڭجياڭ ئۆلكىلىك مېتېرولوگىيە ئىدارىسى » شۋىرغاندىن كۆك رەڭلىك ئاگاھلاندۇرۇش چىقاردى

Figure 2: Example of transferring NE tags from Chinese to Uyghur using NER tools and NMT

2 Related Work

Named Entity Recognition: NER is typically framed as a task of sequence labeling which aims at automatic detection of NEs in free text(Marrero et al., 2013). CRF, SVM, and perceptron models with hand-crafted features are applied in early works (Ratinov and Roth, 2009; Passos et al., 2014; Luo et al., 2015). With the great advantages of deep neural networks, research focuses on the neural network-based methods that need less feature engineering and domain knowledge(Lample et al., 2016; Žukov-Gregorič et al., 2018; Zhou et al., 2019). Collebert (2011) proposed a feed-forward neural network with a fixed-sized window for each word, which failed in considering useful relations between long-distance words. To overcome this limitation, Chiu et al. (2016) presented a bidirectional LSTM-CNNs architecture that automatically detects word and character-level features. Ma et al. (2016) further extended it into bidirectional LSTM-CNNs-CRF architecture, where the CRF module was added to optimize the output label sequence.

Transfer learning for NER: Low-resource languages often suffer from a lack of annotated corpora to estimate high-performing neural network models for many NLP tasks. Transfer learning is an efficient way to bridge the gap across languages. Transfer learning methods for NER can be divided into two types: parallel corpora based and shared representation based transfer. Early works mainly focus on parallel corpora to projecting information from high-resource languages to low-resource languages (Yarowsky et al., 2001; Ehrmann et al., 2011; Wang et al., 2013; Fang and Cohn, 2016; Ni et al., 2017). Chen et al. (2010) and Wang et al. (2013) proposed to jointly identify and align bilingual named entities. Kim el al. (2012), Nothman et al. (2013) and Tsai el al. (2016) using the Wikipedia information to improve low-resource NER. Mayhew et al. (2017) created a cross-language NER system by translating annotated data of high-resource to low-resource which works well for very minimal resource languages. On the other hand, the shared representation methods do not require parallel corpora. Fang et al. (2017) proposed cross-lingual word embeddings to transfer knowledge across resources. Pan et al. (2017) proposes a large-scale cross-lingual named entity dataset which contains 282 languages for evaluation. Yang el al. (2016), Wang et al. (2017), Lin et al. (2018) and Liu et al. (2018) shows that jointly training on multiple tasks or languages helps improve performance. Different from transfer learning methods, multi-task learning aims at improving the performance of all the resources instead of low resource only.

Token Added Machine Translation (TAMT): The researchers proposed TAMT methods to solve the different types of problems. Ugawa et al. (2018) add the entity tags to the source language sentences to disambiguate the multi-meaning entities in the target language. Li et al. (2018) use NE tags to indicate the NE boundary information in the source language sentences to get better customized entity translation. Bai et al. (2018) use some special tokens to mark the segmentation boundary for the slot value in the source sentence and transfer the source language spoken language understanding corpus to the target language.

3 Methodology

3.1 General-Purpose NMT System

Machine translation (MT) translates text sentences from a source language to a target language and the Transformer model is the first NMT model relying entirely on self-attention to compute representations of its input and output without using recurrent neural networks (RNN) or convolutional neural networks (CNN). Our general-purpose NMT system is based on the Transformer model.

The Transformer model is an encoder-decoder structure like most competitive neural sequence transduction models, as shown in Figure 3. The encoder is including three steps, in the first step, the input words are projected into an embedding vector space, position embedding is also added to input vectors to capture the notion of token position within the sequence. The second step is a multi-head self-attention. This is an extension of the previous attention scheme. Instead of using a single attention function, this step computes multiple attention blocks over the source, concatenates them and projects them linearly back onto a space with the original dimensionality. The scaled dot-product attention with different linear projections is computed over attention blocks individually. Finally, a position-wise fully connected feed-forward network is used, which consists of two linear transformations with a ReLU activation.

The decoder works similarity, from left to right with generates one word at a time. It including five steps. The first step: embedding and position encoding, is similar to the encoder. The second step is masked multi-head attention, which masks future words forces to attend only to past words. The third

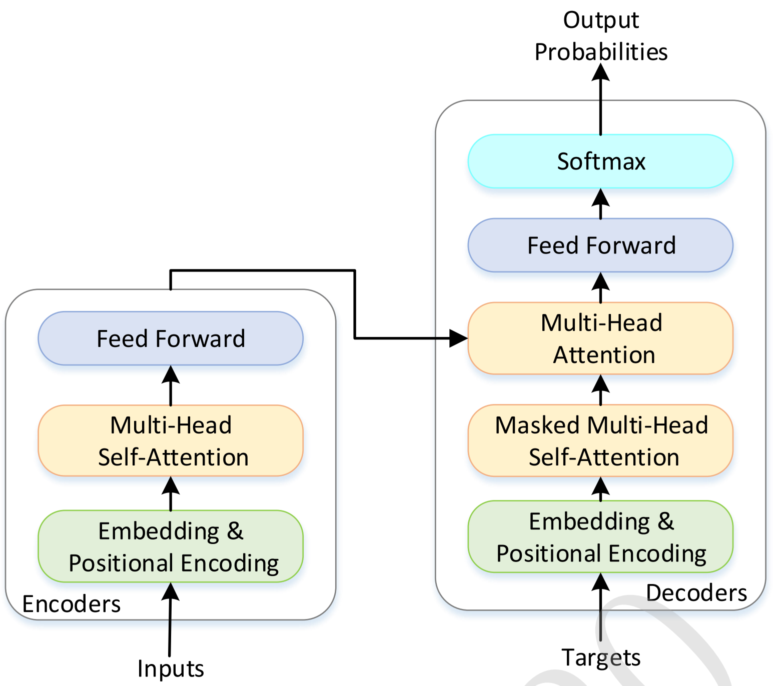

Figure 3: Simplified diagram of the Transformer model

step is a multi-head attention that not only attends to these past words, but also to the final representations generated by the encoder. The fourth step is another feed-forward network. Finally, a softmax layer applied to map target word scores into target word probabilities. More details about the model are found in the original paper (Vaswani et al., 2017).

3.2 Source language Named Entity Tags

We consider three NE classes in this paper (PER, LOC, ORG). For every NE in source sentence, we generate the candidate NE class tags using three types of third-party NER tools: Pyltp [0] from Harbin Institute of Technology, PaddleHub [1] from Baidu and THULAC [2] from Tsinghua University. In order to get the best tags from candidates, we will try two kinds of strategies described as follows:

Single tag combination (STC): Check these tools on a test set to get accuracy for each NE class, then use the highest accuracy tool to get the specific single class tag, such as PER from Pyltp, LOC from PaddleHub and so on.

Multi-tag combination (MTC): For single sentences, tags are comes from all three tools and combine them by following rules:(1) Tag kept for a single NE only if all of the three tags are identical. (2) Tag kept for the longer NE if NE from one tool includes another one. (3) Drop the sentences not satisfy any of the first two rules.

3.3 Token-added Translation

To make the general-purpose NMT aware of NEs, we propose a token added translation approach. This approach uses some special tokens to mark the segmentation boundary for the NE in the source sentence. These special tokens are common in both the source vocabulary and target vocabulary of the general-purpose NMT and their translation is unique and easy to spot. To avoid complexity, we use the same common special tokens for all NEs while keeping order and mark all NEs in the translated target sentences with the original order. For example, punctuation like parentheses and double quotes are good candidates as special tokens. Enclosing NEs in source sentences by these special tokens can help iden-

[0] https://github.com/HIT-SCIR/pyltp
[1] https://github.com/PaddlePaddle/PaddleHub
[2] https://github.com/thunlp/THULAC-Python

tify NE boundaries in the translation outputs. In our example in Figure 2, the special tokens we choose is a pair of Chinese punctuation named title mark (《》), which translated to corresponding Uyghur punctuation («»).

In token-added translation, no additional word alignment process is required. However, such an approach relies heavily on the NMT general training data where the special tokens (e.g. parentheses or double quotation marks) are kept in both source and target data. For different language pairs, different special tokens might be chosen for the best translation quality. Empirically we find that title marks are highly effective for Chinese to Uyghur NE translation.

3.4 NER Model

The hierarchical CRF model consists of three components: a character-level neural network, either an RNN or a CNN, that allows the model to capture subword information, such as morphological variations and capitalization patterns; a word-level neural network, usually an RNN, that consumes word representations and produces context-sensitive hidden representations for each word; and a linear-chain CRF layer that models the dependency between labels and performs inference.

In this paper, we closely follow the architecture proposed by Lample et al. (2016), and use bidirectional LSTMs for both the character level and word level neural networks. Specifically, given an input sequence of words $(w_1, w_2, ..., w_n)$, and each word's corresponding character sequence, the model first produces a representation for each word, x_i, by concatenating its character representation with its word embedding. Subsequently, the word representations of the input sequence $(x_1, x_2, ..., x_n)$ are fed into a word level Bi-LSTM, which models the contextual dependency within each sentence and outputs a sequence of context sensitive hidden representations $(h_1, h_2, ..., h_n)$. A CRF layer is then applied on top of the word level LSTM and takes in as its input the sequence of hidden representations $(h_1, h_2, ..., h_n)$, and defines the joint distribution of all possible output label sequences. The Viterbi algorithm is used during decoding.

4 Experiment

4.1 Data

The CCMT 2017 Chinese-Uyghur corpus [3] is used to train the general-purpose Chinese-Uyghur NMT system and the MSRA dataset from international Chinese language processing Bakeoff 2006 [4] is used to evaluate the performance of Chinese NER tools. As no publicly available test set to evaluate the performance of Uyghur NER, we will randomly choose 2000 sentences from Uyghur named entity relation corpus (Abiderexiti et al., 2016), in which tagged entity tags and relation types, checked the entity tags manually and used 1000 sentences as our Uyghur NER test set and another 1000 sentences as development set. The 1,500,000 Uyghur sentences crawled from the Tianshan website [5] is used to train the Uyghur word embeddings and the BIO tag schema is used where the B, I, O refer to the beginning, inside and outside of an entity, respectively.

4.2 Setup

General-Purpose NMT: We use the Transformer model (Vaswani et al., 2017) implemented in PyTorch in the fairseq-py (Ott et al., 2019) toolkit and all experiments are based on the "base" transformer model. We use word representations of size 512, feed-forward layers with inner dimension 2048, and multi-headed attention with 8 attention heads. We apply dropout with probability 0.3. Models are optimized with Adam using β_1= 0.9, β_2 = 0.98, and ε = 1e-8 . We use the same learning rate schedule as Vaswani et al. (Vaswani et al., 2017) , i.e., the learning rate increases linearly for 4,000 steps to 5e-4 (or 1e-3 in experiments that specify 2x *lr*), after which it is decayed proportionally to the inverse square root of the number of steps. We use label smoothing with 0.1 weight for the uniform prior distribution over the vocabulary. All experiments are run on 2 NVIDIA V100 GPUs interconnected by Infiniband.

[3] http://ee.dlut.edu.cn/CWMT2017/index_en.html

[4] http://sighan.cs.uchicago.edu/bakeoff2006/

[5] http://uy.ts.cn/

NER Model: We use the 300-dimensional word embeddings pretrained by Word2Vec, FastText, and Glove respectively. We set the character embedding size to be 100, character level LSTM hidden size to be 25, and word-level LSTM hidden size to be 100. For OOV words, we initialize an unknown embedding by uniformly sampling from range [-$\sqrt{\frac{3}{emb}}$,+$\sqrt{\frac{3}{emb}}$] where *emb* is the size of embedding, 300 in our case. We train the model for 100 epochs and optimize the parameters by Stochastic Gradient Descent (SGD) with momentum, gradient clipping, and learning rate decay. We set the learning rate (*lr*) and the decay rate (*dr*) as 0.01 and 0.05 respectively. To prevent overfitting, we apply dropout with a rate of 0.5 on outputs of the two Bi-LSTMs.

4.3 Results and Analysis

4.3.1 Comparison of tag combination strategy

1)Result of the STC Strategy

To obtain the accuracy of the three named entity recognition systems for the recognition of each entity type, we conducted experiments on the MSRA data set, and the experimental results are shown in Table 1.

NER system	Entity Type	Accuracy	Recall	F1
PaddleHub	LOC	81.09	66.77	73.24
	PER	83.16	80.08	81.59
	ORG	70.31	61.38	**65.54**
	ALL	79.51	69.86	74.37
Pyltp	LOC	86.26	71.81	**78.38**
	PER	90.73	61.53	73.33
	ORG	82.21	48.61	61.10
	ALL	86.88	63.53	73.40
THULAC	LOC	73.58	65.73	69.43
	PER	86.93	85.25	**86.08**
	ORG	78.06	16.30	26.97
	ALL	79.24	61.32	69.14

Table 1: The results of three Chinese NER system.

The experimental results show that PaddleHub has the best recognition for ORG while Pyltp for LOC and THULAC for PER. Therefore, the results of three NER systems are fused according to the STC strategy, and the fusion results are shown in Table 2. It can be seen that the recognition performance of the single and all entity is higher than the original system.

Strategy	Entity Type	Accuracy	Recall	F1
STC	LOC	90.10	70.56	79.14
	PER	86.93	85.25	86.08
	ORG	70.47	61.31	65.57
	ALL	84.70	73.26	**78.56**

Table 2: The results of the STC.

2)Result of the MTC Strategy

the result of the MTC is strategy shown in Table 3. Comparing the results of Table 2 and Table 3, it can be seen that the STC strategy is better than the MTC strategy for the recognition of Chinese named entities, and the following experiments are based on the STC strategy.

4.3.2 Baseline

To show the effectiveness of the proposed method, a strong baseline system is needed. In this paper, we will gradually explore the impact of different word alignment tools and different word vector models on

Strategy	Entity Type	Accuracy	Recall	F1
MTC	LOC	88.35	67.23	76.37
	PER	81.56	71.89	75.31
	ORG	69.67	54.38	61.08
	ALL	79.86	64.50	71.36

Table 3: The results of MTC.

cross-lingual entity migration, and finally, build a cross-language entity migration baseline system based on the parallel corpus and word alignment tools.

1) Comparison of Word Alignment Tools

word alignment accuracy is very important for word alignment based cross-lingual NER system and GIZA++ (Casacuberta and Vidal, 2007), fast_align (Dyer et al., 2013), and efmaral(Östling and Tiedemann, 2016) are currently popular word alignment tools. We will construct an Uyghur NER system using these three types of word alignment tools with the STC strategy based on the Uyghur-Chinese parallel corpus and The performance is shown in Table 4. It can be seen that efmaral word alignment tool has the best performance for our task.

Tools	Entity Type	Accuracy	Recall	F1
GIZA++	LOC	82.36	43.22	56.69
	PER	95.15	32.24	48.16
	ORG	73.07	41.11	52.62
	ALL	82.04	39.92	**53.71**
fast_align	LOC	80.61	56.36	66.43
	PER	96.93	36.35	52.87
	ORG	65.30	44.25	52.75
	ALL	79.08	48.37	**60.03**
efmaral	LOC	80.17	69.83	74.65
	PER	87.54	40.46	55.34
	ORG	66.88	53.48	59.44
	ALL	77.93	58.44	**66.69**

Table 4: Comparison of three word alignment tools.

2) Comparison of Word Embeddings

Word embeddings can provide rich semantic information and allow the system to better capture the semantic relevance between words. we will use the static word embeddings generated from Word2Vec, Glove, and FastText separately to initialize the network input and explore the effect of different word vectors on Uyghur NER construction. The Experimental results are shown in Table 5 and it can be seen that Word2Vec generated embeddings have good performance for our task.

Word Embedding	Accuracy	Recall	F1
Random	77.93	58.44	66.69
Glove	78.50	61.37	68.89
FastText	78.58	61.50	69.00
Word2Vec	79.17	62.75	**70.01**

Table 5: Comparison of three types of word embedding.

4.3.3 Analysis of Token-added Translation Method

We use the STC strategy to get named entity tags in Chinese and use the proposed token-added translation method to translated the entities to Uyghur to construct tagged NER corpus. Finally, train an Uyghur NER system using this data and the performance is shown in Table 6.

Method	Entity Type	Accuracy	Recall	F1
Token-add translation	LOC	66.45	50.91	57.65
	PER	79.96	69.57	74.41
	ORG	47.28	28.75	35.75
	ALL	66.70	50.33	**57.37**

Table 6: Uyghur NER based on token-added translation method.

From Table 6, it can be seen that the token-add translation method has worse performance compared with baseline. After analyzing the data, we found that only Uyghur stems are included in the special token while most of the affixes appended by the stem are being excluded. As an agglutinative language, Uyghur has rich affixes to express grammatical information in the sentence. For example, as shown in Figure 4, the original Chinese entity "新疆" is included in the Chinese bookmark (《》) and translated to Uyghur by MT, it can be found that the translated Uyghur entity also included in Uyghur bookmark («») while appended affix is excluded.

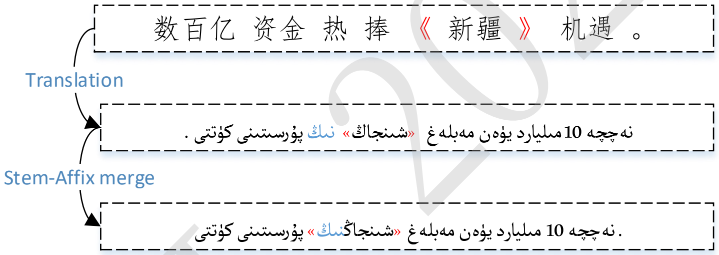
数百亿 资金 热 捧 《 新疆 》 机遇 。
Translation
نەچچە 10 ملیارد یۈەن مەبلەغ «شنجاڭ» نڭ پۈرستنی کۈتتی .
Stem-Affix merge
نەچچە 10 ملیارد یۈەن مەبلەغ «شنجاڭنڭ» پۈرستنی کۈتتی .

Figure 4: The example of entity boundary characters based entity translation

To prevent the problem, we apply a stem-affix merge method for translated Uyghur sentences and merge the stem with the followed word if it is affix. We train a new Uyghur NER system using handled corpus and the result is shown in Table 7. It can be seen that the combination of stem and affixes can effectively avoid the affix as a separate word in the corpus, thereby greatly improving the quality of the corpus and the performance of trained Uyghur NER system significantly, the f1 score is 3.79% higher than the baseline.

Method	Entity Type	Accuracy	Recall	F1
Stem-Affix merged	LOC	78.57	70.91	74.54
	PER	80.78	81.58	81.18
	ORG	67.05	61.32	64.06
	ALL	76.47	71.32	**73.80**

Table 7: Result of Stem-Affix merged method

5 Conclusion

Aiming at the lack of Uyghur named entity recognition training corpus, this paper proposes a cross-language named entity tag transfer method based on general machine translation and entity boundary token. First obtains the named entity tags of Chinese sentences in Chinese-Uyghur parallel corpus through a variety of Chinese named entity recognition tools and uses tag fusion strategies to fuse multi-source tags, then select appropriate special symbols to surround the entities and uses Chinese-Uyghur neural machine translation system to translate the Chinese sentences to Uyghur. Finally, the Uyghur stems and affixes merge method is used to obtain a high-quality Uyghur named entity recognition corpus. The Uyghur NER system trained with this corpus achieved good performance, which was 3.79% points higher than the baseline system.

Acknowledgements

This work is supported in part by A Class Funded Project of the Western Light Talent Training Program of the Chinese Academy of Sciences(2017-XBQNXZ-A-005), NSFC(U1703133), The West Light Foundation of The Chinese Academy of Sciences(Grant No.2019-XBQNXZ-B-008), The National Key R&D Plan(2017YFC0822505-04).

References

Kahaerjiang Abiderexiti, Maihemuti Maimaiti, Tuergen Yibulayin, and Aishan Wumaier. 2016. Annotation schemes for constructing uyghur named entity relation corpus. In *2016 International Conference on Asian Language Processing (IALP)*, pages 103–107. IEEE.

He Bai, Yu Zhou, Jiajun Zhang, Liang Zhao, Mei-Yuh Hwang, and Chengqing Zong. 2018. Source-critical reinforcement learning for transferring spoken language understanding to a new language. *arXiv preprint arXiv:1808.06167*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*.

Emre Cakır and Tuomas Virtanen. 2019. Convolutional recurrent neural networks for rare sound event detection. *Deep Neural Networks for Sound Event Detection*, 12.

Francisco Casacuberta and Enrique Vidal. 2007. Giza++: Training of statistical translation models. *Retrieved October*, 29:2019.

Yufeng Chen, Chengqing Zong, and Keh-Yih Su. 2010. On jointly recognizing and aligning bilingual named entities. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 631–639. Association for Computational Linguistics.

Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. A walk-based model on entity graphs for relation extraction. *arXiv preprint arXiv:1902.07023*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.

Maud Ehrmann, Marco Turchi, and Ralf Steinberger. 2011. Building a multilingual named entity-annotated corpus using annotation projection. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, pages 118–124.

Meng Fang and Trevor Cohn. 2016. Learning when to trust distant supervision: An application to low-resource pos tagging using cross-lingual projection. *arXiv preprint arXiv:1607.01133*.

Meng Fang and Trevor Cohn. 2017. Model transfer for tagging low-resource languages using a bilingual dictionary. *arXiv preprint arXiv:1705.00424*.

Jerry R Hobbs, Douglas Appelt, John Bear, David Israel, Megumi Kameyama, Mark Stickel, and Mabry Tyson. 1997. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. *Finite-state language processing*, pages 383–406.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Lifu Huang, Kyunghyun Cho, Boliang Zhang, Heng Ji, and Kevin Knight. 2018. Multi-lingual common semantic space construction via cluster-consistent word embedding. *arXiv preprint arXiv:1804.07875*.

Sungchul Kim, Kristina Toutanova, and Hwanjo Yu. 2012. Multilingual named entity recognition using parallel data and metadata from wikipedia. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 694–702. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Zhongwei Li, Xuancong Wang, Ai Ti Ai, Eng Siong Chng, and Haizhou Li. 2018. Named-entity tagging and domain adaptation for better customized translation.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 799–809.

Liyuan Liu, Jingbo Shang, Xiang Ren, Frank Fangzheng Xu, Huan Gui, Jian Peng, and Jiawei Han. 2018. Empower sequence labeling with task-aware neural language model. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. 2015. Joint entity recognition and disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 879–888.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Maihemuti Maimaiti, Aishan Wumaier, and Kahaerjiang Abiderexiti. 2018. Construction of uyghur named entity corpus. *Belt & Road: Language Resources and Evaluation*, page 2.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489.

Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. 2017. Cheap translation for cross-lingual named entity recognition. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2536–2545.

Jian Ni, Georgiana Dinu, and Radu Florian. 2017. Weakly supervised cross-lingual named entity recognition via effective annotation and representation projection. *arXiv preprint arXiv:1707.02483*.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence*, 194:151–175.

Robert Östling and Jörg Tiedemann. 2016. Efficient word alignment with markov chain monte carlo. *The Prague Bulletin of Mathematical Linguistics*, 106(1):125–146.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. *arXiv preprint arXiv:1404.5367*.

Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. *arXiv preprint arXiv:1705.00108*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via wikification. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 219–228.

Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. 2018. Neural machine translation incorporating named entity. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3240–3250.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Mengqiu Wang, Wanxiang Che, and Christopher D Manning. 2013. Joint word alignment and bilingual named entity recognition using dual decomposition. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1082.

Dingquan Wang, Nanyun Peng, and Kevin Duh. 2017. A multi-task learning approach to adapting bilingual word embeddings for cross-lingual named entity recognition. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 383–388.

Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *arXiv preprint arXiv:1603.06270*.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Xi Peng, Yang Xiao, and Zhiguo Cao. 2019. Roseq: Robust sequence labeling. *IEEE transactions on neural networks and learning systems*.

Andrej Žukov-Gregorič, Yoram Bachrach, and Sam Coope. 2018. Named entity recognition with parallel recurrent neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 69–74.