Deep Multi-task Learning with Cross Connected Layer for Slot Filling

Junsheng Kong[1], Yi Cai[1(✉)], Da Ren[1], and Zilu Li[2]

[1] South China University of Technology, Guangzhou, China
ycai@scut.edu.cn
[2] Guangzhou Tianhe Foreign Language School, Guangzhou, China

Abstract. Slot filling is a critical subtask of Spoken language understanding(SLU) in task-oriented dialogue systems. This is a common scenario that different slot filling tasks from different but similar domains have overlapped sets of slots (shared slots). In this paper, we propose an effective deep multi-task learning with Cross Connected Layer (CCL) to capture this information. The experiments show that our proposed model outperforms some mainstream baselines on the Chinese E-commerce datasets. The significant improvement in the F1 socre of the shared slots proves that CCL can capture more information about shared slots.

Keywords: Multi-task learning · Slot filling · Shared slots · CCL

1 Introduction

Spoken Language Understanding, which aims to interpret the semantic meanings conveyed by input utterances, is an important component in task-oriented dialog systems. Slot filling, a sub-problem of SLU, extracts semantic constituents by using the words of input utterance to fill in predefined slots in a semantic frame [1].

Slot filling problem can be regarded as a sequence labeling task, which assigns an appropriate semantic label to each word with a given input utterance. State-of-the-art sequence labeling models are typically based on BiLSTM-CRF [2,3]. There are a variety of task-oriented dialog systems of different domains such as air travel [4], computer shopping guide and phone shopping guide.

It is common that there is semantic correspondence between slots defined in different domains. Consider these two sentences:

1. I want to buy a computer which is about $\{_{price\_middle}$ four thousand yuan$\}$ and has a ram of $\{_{ram\_size}$ 8 G$\}$.
2. I plan to get a new mobile phone of $\{_{brand}$ Huawei$\}$, which costs about $\{_{price\_middle}$ 3000 yuan$\}$.

These two sentences respectively come from the domain of computer shopping guide and the domain of phone shopping guide. We can find out that these two sentences have the same slot "price_middle" which means the median of psychological price. We denote the same slot between the slot filling tasks of different domains as shared slot.

Multi-task Learning (MTL) has been applied to various models and problems. But most of the current multitasking learning methods [5] only share parameters at the bottom layer. We believe that this model structure can't make good use of the information of shared slots. To achieve this goal, we propose the CCL to encourage all tasks to learn a common label representation. We combine the multi-task BiLSTM-CRF model and this layer to address for slot filling problem with shared slots.

The main contribution of this work lies on:

1. We propose an original MTL architecture with CCL to capture the information of shared slots. The experiment results show the effectiveness of the CCL.
2. We build three datasets for slot filling tasks on three domains: computer, mobile phone, and camera. These datasets enrich the experimental data in Chinese slot filling field.

The rest of the paper is organized as follows: Section 2 introduces related work. Section 3 describes the details of our method. In Section 4, we illustrate our experiments. Finally, we draw our conclusions in Section 5.

2 Related Work

Our work is in line with existing methods using neural network for slot filling. Slot filling can be treated as a sequence labeling problem. Here, we use the IOB [6] scheme for representing segmentation. In recent years, deep learning approaches have been explored due to its successful application in many NLP tasks. For slot filling problem, deep learning search has started as extensions of DNNs and DBNs [7] and is sometimes merged with CRFs [8]. Especially with the rediscovery of LSTM cells [9] for RNNs, this architecture has started to emerge [10]. Many neural network architectures have been used such as simple RNNs [11, 12], convolutional neural networks (CNNs) [13], LSTMs [14] and variations like encoder-decoder [15] and external memory [16]. In general, these works adopt a BiLSTM as the major labeling architecture to extract various features, then use a CRF layer [2] to model the label dependency.

MTL has attracted increasing attention in both academia and industry recently. By jointly learning across multiple tasks [17], we can improve performance on each task and reduce the need for labeled data. There has been several attempts of using multi-task learning on sequence labeling task [18–21]. Hakkani-Tur et al. [5] proposed a multi-domain SLU model using BiLSTM. They train the multi-domain model by using data from all the domains and let the data from each domain to reinforce each other. Kim et al. proposed neural generalization of the feature augmentation domain adaptation methods [22]. Their model uses an LSTM to captures global patterns by training on data.

3 Model

To better utilize the shared slots' information between multiple tasks, we propose a novel multi-task architecture with CCL (MT-BiLSTM-CRF-CCL). As shown in Fig.1, the architecture contains four layers: (1) Embedding Layer, (2) Shared BiLSTM Layer, (3) Cross Connected Layer, (4) Task-oriented CRF Output Layer.

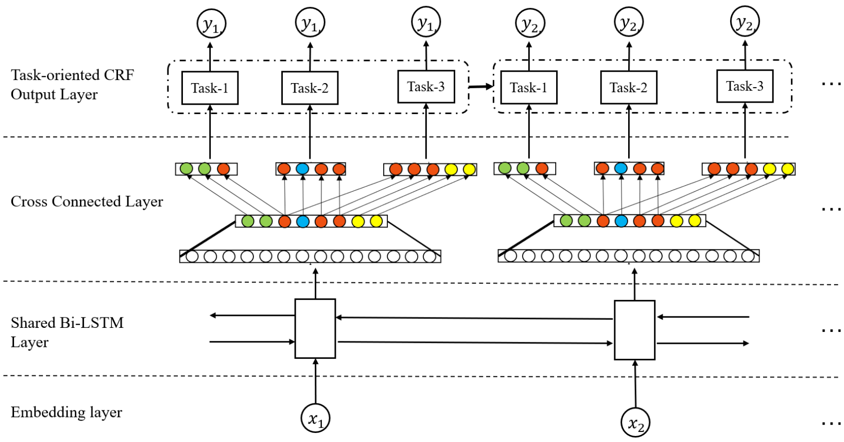

Fig. 1: The architecture of our proposed model

When learning multiple task $t = \{1, 2, \ldots, T\}$, we denote the task-specific label set as L_t. Each task shares the same BiLSTM layer to get the representations, partially share the last fully connected layer, and has their own CRF output layer. In the training procedure, each task updates the whole model's parameters one by one in each training epoch. Different from previous multi-task work, we propose a novel CCL for multi-task learning to capture the information of shared slots. We will introduce the details of these four layers as follows.

3.1 Embedding Layer

In this layer, we take the characters as the input of our model. It has been shown that character-based methods outperform word-based methods for Chinese sequence tagging problem [23, 24]. In the character sentence $S = \{c_1, c_2, \ldots, c_n\}$, each character c_i is represented using

$$x_i^c = e^c(c_i) \tag{1}$$

where e^c denotes a character embedding looking table.

3.2 Shared BiLSTM Layer

Long short-term memory(LSTM) [9] is a recurrent neural network (RNN) that models interactions between input vectors and hidden layers. Since most sequence tagging tasks benefit from both historical and future context information of the words when deciding on the label for a given token, most LSTM sequence taggers use the bi-directional model. The BiLSTM uses two LSTMs to enables the hidden states to capture both historical and future context information of the words.

We build a shared BiLSTM layer for several tasks to get the representation of the input sentences. The shared BiLSTM layer is trained with the feedback by all the tasks. Mathematically, the input of this BiLSTM layer is the output of the embedding layer $X = \{x_1^c, x_2^c, \ldots, x_n^c\}$. The output of BiLSTM layer is a sequence of the hidden states for each input word, denoted as $H = (h_1, h_2, \ldots, h_n)$. Each final hidden state is the concatenation of the forward $\overrightarrow{h}$ and backward $\overleftarrow{h}$ hidden states. We view BiLSTM as a function $BiLSTM(x_i)$:

$$\overrightarrow{h}_i = LSTM(x_i, \overrightarrow{h}_{i-1}), \overleftarrow{h}_i = LSTM(x_i, \overleftarrow{h}_{i-1}) \tag{2}$$

$$BiLSTM(x_i) = h_i = \overrightarrow{h}_i(x_i) \oplus \overleftarrow{h}_i(x_i) \tag{3}$$

In addition, we stack multiple BiLSTMs to make the model deeper, in which the output h_i^l of layer l becomes the input of layer $l+1$, e.g. $h_i^{l+1} = BiLSTM^{l+1}(h_i^l)$

3.3 Cross Connected Layer

In previous multi-task work on multi-domain, each task makes a task-oriented fully connected layer or a task-oriented CRF as the output layer. The information between multiple tasks is shared at the BiLSTM layer. But this multi-task model only shares parameters at the bottom layer, it is difficult for multi-task model to obtain information between slot labels. Therefore, in order to better extract the information of shared slots, we propose a novel CCL to encourage the multi-task model to learn a shared label representation between different tasks. CCL has two processes: union process and separate process.

Union process: this process transforms the hidden states h_i into the union label representation D_i on the union slots L_u.
The union slots L_u is

$$L_u = \bigcup_t L_t. \tag{4}$$

The shared label representation D_i is

$$D_i = h_i \ W_{fc}. \tag{5}$$

Here, the dimensionality of W_{fc} is $s \times l$ where s is the hidden size and l is the size of union slots L_u.

The parameters will be updated by all of shared slots' training data of all tasks, and this make our model get better use of the information of shared slots.

Separate process: this process convert the shared label representation D_i to the task-oriented label representation $D_{(i,t)}$ for task t. For each task-oriented label representation $D_{(i,t)}$, it only use the label representation of corresponding slots. This process avoids the problem caused by that we mix all slots into one label representation. The procedure of the separate algorithm is summarized as Algorithm 1.

Algorithm 1: Separate slots

```
for t in 1, 2, ..., T do
    for (index, slot) in L_u do
        if slot in L_t then
            D_(i,t).append(D_i[index])
        end
    end
end
```

Among the algorithm, the "index" and "slot" here represent the subscript and label value of the L_u respectively.

3.4 Task-oriented CRF Output Layer

Label dependencies are crucial for sequence labeling tasks. For example, in the task with IOB annotation, it is not only meaningless but illegal to annotate I-price_middle after B-brand (i.e., mixing the price and the brand). Therefore, jointly decoding a chain of labels can ensure the resulting label sequence to be meaningful. CRF has been included in most state-of-the-art models to capture such information and further avoid generating illegal annotations [25]. Consequently, we build a task-oriented standard CRF layer upon the CCL layer for all tasks.

The CRF layer takes the output of the CCL $D_{(1,t)}, D_{(2,t)}, \ldots, D_{(n,t)}$ as input. For task t, the probability of a label sequence $y_t = l_1, l_2, \ldots, l_n$ is

$$P(y_t|s) = \frac{exp(\sum_i (W_{CRF-t}^{l_i} h_{(i,t)} + b_{CRF-t}^{(l_{(i-1,t)}, l_{(i,t)})}))}{\sum_{y'} (exp(\sum_i (W_{CRF-t}^{l_i} h_{(i,t)} + b_{CRF-t}^{(l_{(i-1,t)}, l_{(i,t)})})))} \tag{6}$$

where y' represents an arbitary label sequence, $W_{CRF-t}^{l_i}$ is a model parameter specific to l_i of task t, and $b_{CRF-t}^{(l_{(i-1,t)}, l_{(i,t)})}$ is a bias specific to $l_{(i-1,t)}$ and $l_{(i,t)}$.

We predict the output sequence by using the Viterbi algorithm [26] to find the highest scored label sentence. Given a set of training data $\{(s_i, y_i)|_{i=1}^{N}\}$, the loss function is

$$L = -\sum_{l=1}^{N} log(P(y_i|s_i)). \tag{7}$$

4 Experiments

4.1 Datasets

We evaluate the proposed model on the datasets across multiple domains: E-commerce Computer, E-commerce Camera, E-commerce Phone. These datasets are obtained from the websites of the camera, computer and mobile phone. Then, we manually filter and tag the data to get the final datasets. These datasets are divided into three parts: train set, development set and test set. The vocabulary size of the dataset is 1189. The Table.1 shows the statistics of these datasets. These datasets are available online[3].

Table 1: The statistics of the datasets

Dataset	**Train**	**Dev**	**Test**	**Label num**	**Avg. length**
E-commerce Computer	6145	1087	1113	47	17.20
E-commerce Camera	3408	522	521	25	21.84
E-commerce Phone	3455	626	616	37	19.00

4.2 Experimental Setting

We use the 100-dimensional character embedding from Wikimedia documents trained by word2vec. For the shared BiLSTM Layer, we use a 2-layers BiLSTM. The hidden size of BiLSTM is set to be 100. The training procedure consists of two stages: joint training and fine-tuning. We use the Adam optimizer to train our architecture. The batch size 2 and 32 are applied in joint training stage and fine-tuning stage, respectively. The dropout rate is set to be 0.25.

In this experiment, we use several mainstream models including BiLSTM, BiLSTM-CRF, MT-BiLSTM, MT-BiLSTM-CRF. For the single task, we use a BiLSTM and a fully connected layer to form BiLSTM model and apply a CRF layer upon the BiLSTM layer to get BiLSTM-CRF model. For multi-task model (MT-BiLSTM, MT-BiLSTM-CRF), we share the embedding layer and the first LSTM layer.

[3] https://github.com/JansonKong/Deep-Multi-task-Learning-with-Cross-Connected-Layer-for-Slot-Filling

4.3 Experimental Results

We compare the performance of our models to all four baselines. The comparison is conducted on three datasets: E-commerce camera dataset, E-commerce computer dataset, E-commerce phone dataset. The evaluation measures are widely-used Precision, Recall and F1-Measure in information retrieval area. The results are shown on Table.2.

Table 2: Experiment results on three datasets

Model	camera			computer			phone		
	P	R	F1	P	R	F1	P	R	F1
BiLSTM	0.878	0.904	0.8908	0.837	0.823	0.8295	0.812	0.797	0.8045
BiLSTM-CRF	0.904	0.910	0.9071	0.896	0.864	0.8800	0.863	0.863	0.8677
MT-BiLSTM	0.909	0.910	0.9092	0.870	0.837	0.8534	0.892	0.902	0.8969
MT-BiLSTM-CRF	0.916	0.919	0.9175	0.905	0.891	0.8981	0.907	0.910	0.9086
MT-BiLSTM-CCL	0.944	**0.952**	0.9480	0.908	0.908	0.9078	0.893	0.914	0.9033
MT-BiLSTM-CRF-CCL	**0.957**	0.951	**0.9541**	**0.931**	**0.924**	**0.9274**	**0.916**	**0.926**	**0.9207**

From the results, on the the E-commerce camera dataset, single-task BiLSTM model obtains a F1-score of 89.08% and the single-task BiLSTM-CRF model obtains a F1-score of 90.71%. Compared with single-task models, the multi-task models (MT-BiLSTM, MT-BiLSTM-CRF, MT-BiLSTM-CCL, MT-BiLSTM-CRF-CCL) get better results which prove that multi-task models work better on these tasks. The results present that our MT-BiLSTM-CRF-CCL model outperform all the neural network baselines (p-value$<$0.005). Compared with the strong baseline MT-BiLSTM-CRF model (p-value$<$0.005), MT-BiLSTM-CRF-CCL model increases respectively by 3.66%, 2.93% and 1.21% in the F1 scores on these datasets.

In order to verify the effectiveness of CCL, We also combine MT-BiLSTM model with CCL to get MT-BiLSTM-CCL model. The results show that the MT-BiLSTM-CCL model has made a huge improvement in the F1 score compared with MT-BiLSTM model.This result proves the effectiveness of the CCL.

To verify that CCL layer can effectively utilize the information of shared slots, we calculate the F1 values of shared slots and non-shared slot respectively. The results are showed on Fig 2. Compared with the other two models(BiLSTM-CRF, MT-BiLSTM-CRF), we identify that our models with CCL have a similar F1 score on non-shared slots. However, with CCL, both MT-BiLSTM-CRF-CCL and MT-BiLSTM-CCL achieves a significant improvement of F1 score on non-shared slots. This strongly demonstrates that CCL can effectively utilize the information of shared slots from multiple datasets.

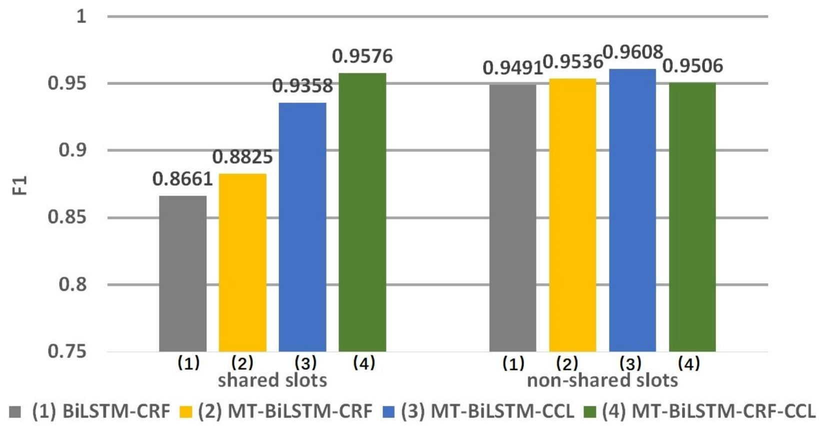

Fig. 2: F1 scores for shared slots and non-shared slots on the E-commerce camera dataset

5 Conclusion

In this paper, we focus on the phenomenon of shared slots between multiple slot filling tasks. In response to this phenomenon, we have proposed a deep multi-task architecture with CCL to capture the information of shared slots. To verify the validity of our approach, we build datasets in three domains. The experimental results show that our model can improve the performance of different slot filling tasks with shared slots. In addition, the proposed method is directly effective and can be easily applied to a similar multi-task model. In the future, we will try to use the CCL to solve other similar problems like text classification tasks with same labels.

Acknowledgment

This work presented in this paper is partially supported by the Fundamental Research Funds for the Central Universities, SCUT (Nos. 2017ZD048, D2182480), the Tiptop Scientific and Technical Innovative Youth Talents of Guangdong special support program (No.2015TQ01X633), the Science and Technology Planning Project of Guangdong Province (No.2017B050506004), the Science and Technology Program of Guangzhou (Nos. 201704030076, 201802010027).

References

1. Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tür, D.Z., He, X., Heck, L.P., Tür, G., Yu, D., Zweig, G.: Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Trans. Audio, Speech & Language Processing

23(3), 530–539 (2015). https://doi.org/10.1109/TASLP.2014.2383614, https://doi.org/10.1109/TASLP.2014.2383614
2. Huang, Z., Xu, W., Yu, K.: Bidirectional LSTM-CRF models for sequence tagging. CoRR **abs/1508.01991** (2015), http://arxiv.org/abs/1508.01991
3. Reimers, N., Gurevych, I.: Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. CoRR **abs/1707.06799** (2017), http://arxiv.org/abs/1707.06799
4. Price, P.J.: Evaluation of spoken language systems: the ATIS domain. In: Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, USA, June 24-27, 1990 (1990), https://aclanthology.info/papers/H90-1020/h90-1020
5. Hakkani-Tür, D., Tür, G., Çelikyilmaz, A., Chen, Y., Gao, J., Deng, L., Wang, Y.: Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In: Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016. pp. 715–719 (2016). https://doi.org/10.21437/Interspeech.2016-402, https://doi.org/10.21437/Interspeech.2016-402
6. Ramshaw, L.A., Marcus, M.: Text chunking using transformation-based learning. In: Third Workshop on Very Large Corpora, VLC@ACL 1995, Cambridge, Massachusetts, USA, June 30, 1995 (1995), https://aclanthology.info/papers/W95-0107/w95-0107
7. Deoras, A., Sarikaya, R.: Deep belief network based semantic taggers for spoken language understanding. In: INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013. pp. 2713–2717 (2013), http://www.isca-speech.org/archive/interspeech_2013/i13_2713.html
8. Xu, P., Sarikaya, R.: Convolutional neural network based triangular CRF for joint intent detection and slot filling. In: 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, December 8-12, 2013. pp. 78–83 (2013). https://doi.org/10.1109/ASRU.2013.6707709, https://doi.org/10.1109/ASRU.2013.6707709
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735
10. Yao, K., Peng, B., Zhang, Y., Yu, D., Zweig, G., Shi, Y.: Spoken language understanding using long short-term memory neural networks. In: 2014 IEEE Spoken Language Technology Workshop, SLT 2014, South Lake Tahoe, NV, USA, December 7-10, 2014. pp. 189–194 (2014). https://doi.org/10.1109/SLT.2014.7078572, https://doi.org/10.1109/SLT.2014.7078572
11. Yao, K., Zweig, G., Hwang, M.Y., Shi, Y., Yu, D.: Recurrent neural networks for language understanding. In: Interspeech. pp. 2524–2528 (2013)
12. Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al.: Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Transactions on Audio, Speech, and Language Processing **23**(3), 530–539 (2014)
13. Xu, P., Sarikaya, R.: Convolutional neural network based triangular crf for joint intent detection and slot filling. In: 2013 IEEE Workshop on Automatic Speech Recognition and Understanding. pp. 78–83. IEEE (2013)
14. Yao, K., Peng, B., Zhang, Y., Yu, D., Zweig, G., Shi, Y.: Spoken language understanding using long short-term memory neural networks. In: 2014 IEEE Spoken Language Technology Workshop (SLT). pp. 189–194. IEEE (2014)

15. Zhai, F., Potdar, S., Xiang, B., Zhou, B.: Neural models for sequence chunking. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA. pp. 3365–3371 (2017), http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14776
16. Peng, B., Yao, K., Jing, L., Wong, K.: Recurrent neural networks with external memory for spoken language understanding. In: Natural Language Processing and Chinese Computing - 4th CCF Conference, NLPCC 2015, Nanchang, China, October 9-13, 2015, Proceedings. pp. 25–35 (2015). https://doi.org/10.1007/978-3-319-25207-0_3, https://doi.org/10.1007/978-3-319-25207-0_3
17. Caruana, R.: Multitask learning. Machine Learning **28**(1), 41–75 (1997). https://doi.org/10.1023/A:1007379606734, https://doi.org/10.1023/A:1007379606734
18. Peng, N., Dredze, M.: Improving named entity recognition for chinese social media with word segmentation representation learning. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers (2016), http://aclweb.org/anthology/P/P16/P16-2025.pdf
19. Peng, N., Dredze, M.: Multi-task domain adaptation for sequence tagging. In: Proceedings of the 2nd Workshop on Representation Learning for NLP, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017. pp. 91–100 (2017), https://aclanthology.info/papers/W17-2612/w17-2612
20. Yang, Z., Salakhutdinov, R., Cohen, W.W.: Transfer learning for sequence tagging with hierarchical recurrent networks. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings (2017), https://openreview.net/forum?id=ByxpMd9lx
21. Zhao, X., Haihong, E., Song, M.: A joint model based on cnn-lstms in dialogue understanding. In: 2018 International Conference on Information Systems and Computer Aided Education (ICISCAE). pp. 471–475. IEEE (2018)
22. Kim, Y., Stratos, K., Sarikaya, R.: Frustratingly easy neural domain adaptation. In: COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan. pp. 387–396 (2016), http://aclweb.org/anthology/C/C16/C16-1038.pdf
23. Liu, Z., Zhu, C., Zhao, T.: Chinese named entity recognition with a sequence labeling approach: Based on characters, or based on words? In: Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence, 6th International Conference on Intelligent Computing, ICIC 2010, Changsha, China, August 18-21, 2010. Proceedings. pp. 634–640 (2010). https://doi.org/10.1007/978-3-642-14932-0_78, https://doi.org/10.1007/978-3-642-14932-0_78
24. Li, H., Hagiwara, M., Li, Q., Ji, H.: Comparison of the impact of word segmentation on name tagging for chinese and japanese. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014. pp. 2532–2536 (2014), http://www.lrec-conf.org/proceedings/lrec2014/summaries/358.html
25. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001. pp. 282–289 (2001)
26. Forney, G.D.: The viterbi algorithm. Proceedings of the IEEE **61**(3), 268–278 (1973)