Domain-aware Neural Model for Sequence Labeling using Joint Learning

Heng Huang
Alibaba Group
Hangzhou, Zhejiang, China
streakly@gmail.com

Yuliang Yan
Alibaba Group
Hangzhou, Zhejiang, China
yyl8781697@live.com

Xiaozhong Liu*
Indiana University Bloomington
Bloomington, Indiana, USA
liu237@indiana.edu

ABSTRACT

Recently, scholars have demonstrated empirical successes of deep learning in sequence labeling, and most of the prior works focused on the word representation inside the target sentence. Unfortunately, the global information, e.g., domain information of the target document, were ignored in the previous studies. In this paper, we propose an innovative joint learning neural network which can encapsulate the global domain knowledge and the local sentence/token information to enhance the sequence labeling model. Unlike existing studies, the proposed method employs domain labeling output as a latent evidence to facilitate tagging model and such joint embedding information is generated by an enhanced highway network. Meanwhile, a redesigned CRF layer is deployed to bridge the 'local output labels' and 'global domain information'. Various kinds of information can iteratively contribute to each other, and moreover, domain knowledge can be learnt in either supervised or unsupervised environment via the new model. Experiment with multiple data sets shows that the proposed algorithm outperforms classical and most recent state-of-the-art labeling methods.

ACM Reference Format:
Heng Huang, Yuliang Yan, and Xiaozhong Liu. 2019. Domain-aware Neural Model for Sequence Labeling using Joint Learning. In *Proceedings of the 2019 World Wide Web Conference (WWW '19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313566

1 INTRODUCTION

Sequence labeling, including tasks like segmentation, named entity recognition, chunking, etc., is an essential stage for a number of downstream nature language processing applications, e.g., natural language understanding and semantic role labeling. Recently, there have been huge improvements in sequence labeling tasks and these breakthroughs were largely fueled by recent advances in deep learning models. For NLP tasks, most of the newly explored methods focused on characterizing (local) word-level or character-level embedding representation, which uncover the important latent semantic information inside the document for sequence labeling. However, for all of those works [2, 15, 17, 29], the global information, e.g., domain/category information, was ignored for model generation. Meanwhile, experience from other NLP tasks, e.g., question answering and semantic search, tells that domain information, when combining with local features, can be potentially important to further enhance the algorithm performance.

Take the following sentences as an example [Domain: Travel] *Big Apple could be a good choice for all-inclusive vacations*, [Domain: Technology] *In the past black friday, a big Apple sale event was first accounted by...*, the word pair **big apple** plays different roles in these sentences, and domain information could be useful to discover such variation. From sequence labeling viewpoint, "*big apple*" could be labeled differently (e.g., named entity or not) in the different contexts, and the global domain knowledge may contribute to the target named entity recognition model.

In this paper, in order to address this problem, we propose a novel domain-aware sequence labeling neural model, which encapsulates both global and local information by learning an independent joint embedding in each possible domain. The domain features can be characterized in either supervised or unsupervised environment. Moreover, a redesigned CRF layer is utilized to estimate output labels probability based on the statistic distribution which is generated by the joint embeddings learned from the last iteration.

The contribution of this paper contains three aspects. Firstly, to the best of our knowledge, this study is the first attempt to explore the domain knowledge in the neural model for sequence labeling tasks, and unlike prior methods, the latent global information is used as a distributional component in a joint learning framework. Secondly, a two level Bi-LSTM with an innovative highway layer is proposed to enable communications between layers and to distinguish the joint embedding from the direct concatenation of contextual and domain representations. Last but not least, we have verified the proposed method on three open corpora with domain labels for the tasks named entity recognition (NER) and segmentation, and another standard dataset without domain labels associated with the chunking task. The experiments show the superiority of the new model compared with a number of classical and most recent state-of-the-art baselines. Meanwhile, experiment results support our initial hypothesis that domain information can be important for sequence labeling tasks.

2 RELATED WORK

For sequence labeling tasks, classic linear statistical methods aim to maximize the joint probability over the linguistic observation data for label generation by leveraging the Markov approach, such like hidden Markov models [22], maximum entropy Markov models [19] and conditional random fields [14]. These methods have been successfully applied in various kinds of NLP tasks. While these models rely heavily on the hand-crafted features, and the long-range

[*] Corresponding Author

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313566

dependency restricts the algorithm performance. In order to break this boundary, in the past few years, neural network approach, e.g., convolutional neural networks (CNN) and recurrent neural network (RNN) approaches, along with the word embedding features were employed as an important alternative for sequence labeling. Collobert et al. [6], for instance, utilized feed-forward neural network to learn the word representations from unlabeled data instead of man-made features and used CNN to extract high-level features. In addition, several models based on RNN [7] and its variation long short-term memory (LSTM) [10] or gated recurrent unit (GRU) [3], were proposed to capture the long-range dependency of the sequential data. More recently, several RNN-like models were proposed for sequence tagging, e.g., deep RNN for speech recognition [8], encoder-decoder via LSTM cell for sequence-to-sequence learning [26], etc. Recent studies also showed the benefits of merging the neural networks and CRF because the representative features obtained by the RNNs exactly fit the need of CRF. For example, Huang et al. [11] combined bidirectional LSTM (Bi-LSTM) with CRF layer and Ma [17] added CNN on Bi-LSTM-CRF for tagging sentences. Furthermore, language models were introduced for calculating the probability of context [16, 21, 23], which produced promising results in the tasks like chunking, named entity recognition, part-of-speech tagging and so on. However, to the best of our knowledge, all the existing approaches only focused on the local information and the global domain potentials were ignored.

Experience from prior studies shows that global information, e.g., domain knowledge, can be potentially important for a number of text mining tasks, e.g., information extraction [1], information retrieval [27, 28], and information summarization [4]. However, most of the previous works employed global information as a kind of simple feature or side-information for model generation. More sophisticated investigations are quite sparse in the deep learning community.

To take advantage of the domain knowledge, in this paper, we propose an innovative domain-aware sequence labeling model. Unlike prior methods, the new model encapsulates both local contextual information and global domain knowledge into a joint learning framework. For each possible domain, the proposed algorithm extracts independent joint representations of context and domain information via a multi-channel highway layer which integrates the outputs of two-level Bi-LSTM, and then estimates output labels probability using a redesigned CRF layer based on the domain distribution.

3 JOINT LEARNING FOR SEQUENCE LABELING WITH DOMAIN KNOWLEDGE

3.1 Problem Definition

To introduce global domain features, we tailor the original linear-CRF by adding a new vertex d that represents the domain knowledge of input sequence X, which is illustrated as the new probabilistic graphical model as Fig.1. Based on the Bayesian chain-rule in DAG [18, 24] the expected output labels Y^* should has a maximum marginal probability as follow:

$$Y^* = \underset{Y}{\operatorname{argmax}} \sum_{i=1}^{k} P(Y|X, d_i)P(d_i|X) \tag{1}$$

$$Y = (y_1, y_2, \cdots y_m)$$

where d_i is the ith domain and $i \in \{1, 2, \cdots k\}$. Following this model and the latent variable d_i, the label sequence depends on both input data X (local features) and domain knowledge d (global features) as an integration of all the distributional components for k domains. Consequently, we build up a joint learning framework to estimate the first factor $P(Y|X, d_i)$ (the output label sequence probability given X in the target domain d_i) and the second $P(d_i|X)$ (the probability that input sequence X falls into the domain d_i) iteratively via extracting joint embedding information for each (X, d_i).

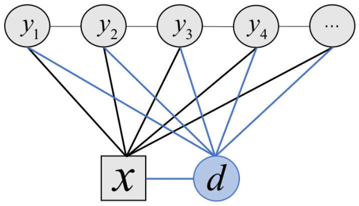

Figure 1: The new DAG for modeling sequence labeling problem based on CRF, in which the vertex d denotes the variant of domain label.

3.2 Proposed model

Based on the proposed model framework, the output label sequence is determined by both joint embedding information and domain distribution. The former factor heavily can be depend on the latter. Hence we propose a joint learning framework illustrated as Fig.2 that contains two parts. The sequence labeling part applies an innovative domain-aware CRF layer to estimate the output label probability via joint representations of domain and context through a joint representation extractor. The other, domain knowledge extractor, is a multi-class classifier for calculating domain distribution of the input sequence. Each token, in the input sequence, is represented as a concatenation of word embedding and character-level embedding, and then passed through two extractors respectively to obtain the joint representation and domain distribution. Finally, the domain-aware CRF layer calculates the probability of the label sequence by using both joint features and the domain knowledge. The two parts in the framework can be trained iteratively by forward pass and back-propagation through time (BPTT) while contributing to each other.

3.3 Joint representation extractor

In the proposed model, in order to learn the joint representation based on context and domain embedding information, a two-level depth bidirectional LSTM and an enhanced highway layer are implemented as shown in Fig.3, where x is the inputs and o denotes the outputs of LSTM cell with ← or → as the forward or backward directions. v_i is the ith domain vector that is initialized randomly, and $s_.^i$ describes the joint representation for the token with the ith domain knowledge.

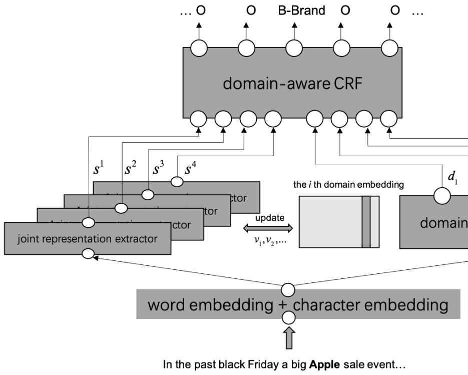
... O
O
B-Brand
O
O ...
domain-aware CRF
s^1
s^2
s^3
s^4
the i th domain embedding
d_1
d_2
d_3
d_4
update
$v_1, v_2, \ldots$
joint representation extractor
domain knowledge extractor
word embedding + character embedding
In the past black Friday a big **Apple** sale event...

Figure 2: The overview of the proposed model. Domain knowledge extractor generates the global domain distribution and the joint representation extractor learns the joint embedding information. Domain-aware CRF layer calculates the output label probability based on the two modules.

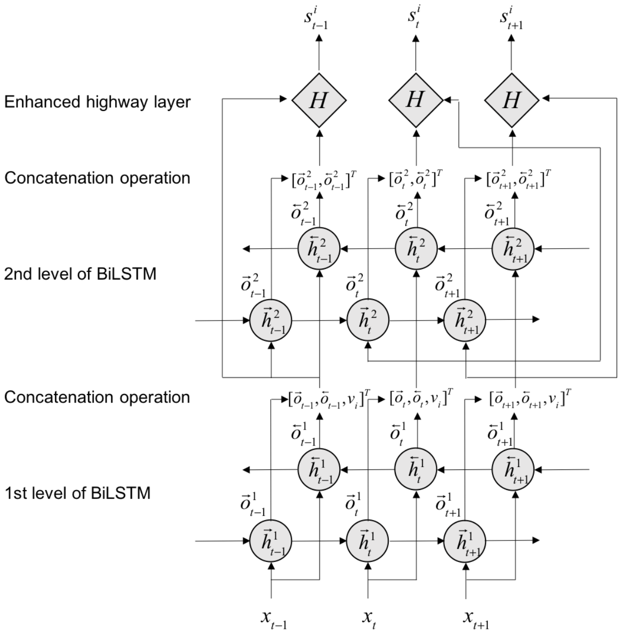
s^i_{t-1}
s^i_t
s^i_{t+1}
Enhanced highway layer
H
H
H
Concatenation operation
$[\vec{o}^2_{t-1}, \overleftarrow{o}^2_{t-1}]^T$
$[\vec{o}^2_t, \overleftarrow{o}^2_t]^T$
$[\vec{o}^2_{t+1}, \overleftarrow{o}^2_{t+1}]^T$
2nd level of BiLSTM
Concatenation operation
$[\vec{o}_{t-1}, \overleftarrow{o}_{t-1}, v_i]^T$
$[\vec{o}_t, \overleftarrow{o}_t, v_i]^T$
$[\vec{o}_{t+1}, \overleftarrow{o}_{t+1}, v_i]^T$
1st level of BiLSTM
x_{t-1}
x_t
x_{t+1}

Figure 3: For the ith domain, we employ two-level Bi-LSTM to embed both context and domain features. The novel multi-channel highway layer balances the contextual and domain representations as the joint representation.

The deep Bi-LSTMs extract the contextual representation and domain embedding information respectively utilizing each level of Bi-LSTMs. Next, for adaptively learning a joint representation from context and domain embeddings, an enhanced highway layer is deployed whose cell has multiple gates to balance the effects of multi-channel inputs. Note that this highway design is quite different from the original one [25]. The multi-channel highway layer allows the networks rationally copy and transform multiple inputs and then output a joint embedding representation. The function of multi-channel highway cell can be formalized as follows:

$$\begin{aligned} s^i_t = \sum_c \lambda_c (F_c(\alpha^i_{c,t}, W^F_c) \cdot T(\alpha^i_{c,t}, W^T_c) \\ + \alpha^i_{c,t} \cdot (1 - T(\alpha^i_{c,t}, W^T_c))) \\ \alpha^i_{c,t} \in \{[\overrightarrow{o}_t, \overleftarrow{o}_t, v_i]^T, [\overrightarrow{o}^2_t, \overleftarrow{o}^2_t]^T\} \\ s.t. \quad \sum_c \lambda_c = 1 \end{aligned} \tag{2}$$

where $\alpha^i_{c,t}$ is the output vector (hidden states) of cth layer of deep Bi-LSTMs at t-step in ith domain and $\cdot$ is element-wise product. $F_c(\cdot, \cdot)$ is the activation function of cth gate and $T(\cdot, \cdot)$ denotes the transition function that determines the proportion of transformed and original input vector α. $W^{\cdot}_{\cdot}$ is the weights for each function that can be learned automatically with the subscript for channel and superscript for function. Note that, for scale invariance, the transformation of the multi-channel highway cell subjects to the criterion that the summation of each weight λ_c equals to 1.

3.4 Domain knowledge extractor

Domain knowledge extractor is essentially a supervised multi-class classifier in this proposed framework. We build up bidirectional LSTM networks and concatenate the final states of the forward and backward networks to optimize and obtain the global features. A softmax layer is deployed to estimate the likelihood that the input sentence should be classified into the target domain. The detailed domain knowledge extractor is depicted in Fig.(4), where x is the input sequence with length n, and d denotes the domain distribution conditioned on input sequence x. For generality, this model could be estimated by MLE and the loss based on softmax function is formularized as follows:

$$J(\theta) = -\frac{1}{m} \sum_{i=1,j=1}^{m,k} 1\{y_i = d_j\} \log \frac{e^{\theta_j^T f_i}}{\sum_{l=1}^k e^{\theta_l^T f_i}} \tag{3}$$

where $1\{\cdot\}$ is indicator function equals to 1 when the ith output y_i matches the ground-truth label d_j. Note that f_i is the concatenation of final states $[\overrightarrow{f}_i, \overleftarrow{f}_i]$ for the ith sentence and θ is the parameters.

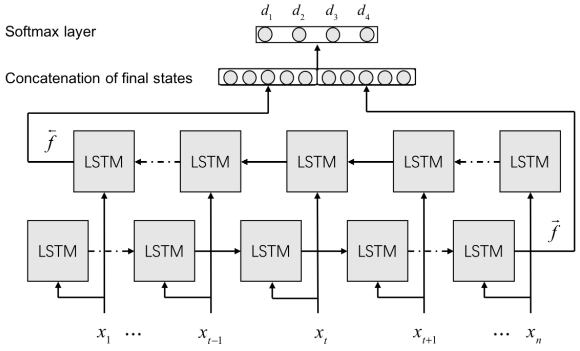

Figure 4: We use the final states of Bi-LSTM for generating the domain distribution.

To generalize the model, the proposed algorithm is able to work in two different scenarios: training samples with or without domain labels. For the latter one, unsupervised clustering algorithms, like K-means, can be cast as a pre-processing and then each training sample can be labeled with a cluster index (pseudo domain label). Finally, we train the model as a supervised learning process.

3.5 Domain-aware CRF

Based on equation (1) and fundamental theorem of random fields [9], the conditional probability can be rewritten as the joint distribution of the minimal cliques:

$$\begin{aligned} P(Y|X) &= \sum_d P(d|X) \frac{\exp(E(y,x,d))}{Z(X,d)} \\ E(y,x,d) &= \sum_k W_k \sum_t f_k(y_{t-1}, y_t, x_t, t, d) \\ Z(X,d) &= \sum_y \exp(\sum_k W_k F_k(y,x,d)) \end{aligned}$$

where $f_k(\cdot)$ is the kth feature function within W_k as the parameter vector. x_t and y_t denote respectively the tth input token and output label subscripted with t as the position. d denotes domain distribution and $Z(\cdot)$ is the normalization which accumulates over all possible label sequences in the finite label set $\{y_l\}$. Note that the feature function $f_k(\cdot)$ consists of two parts: one is to calculate the transition probability from last state to the current and the other represents the correlation between labels and input tokens. Hence, the energy function $E_k(\cdot)$ has the form as the following:

$$\begin{aligned} E(y,x,d) = &\sum_k U_k \sum_t \lambda_k(y_{t-1}, y_t, x_t, t, d) \\ &+ \sum_k V_k \sum_t \mu_k(y_t, x_t, t, d) \end{aligned}$$

whose form is similar to recurrent neural networks. From this perspective, the redesigned domain-aware CRF block is depicted as Fig.5, in which the hidden state h_t is an integration of the probabilistic components in the different domains and d_i denotes the probability corresponding to the input sentence dropping in the ith domain. s_t^i is the jointly embedding vector in the ith domain produced by joint representation extractor. Note that each block C in domain-aware CRF layer is implemented based on forward-backward algorithm using RNN cell [5].

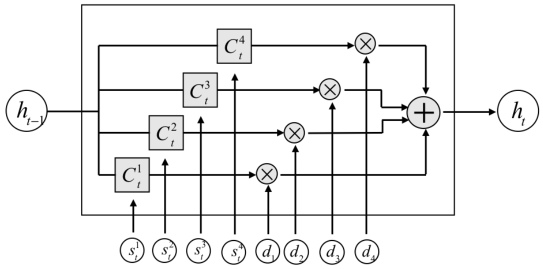

Figure 5: The domain-aware CRF block calculates the marginal probability based on equation (1).

As a discriminative model, the domain-aware CRF can be trained by maximum likelihood estimation (MLE), whose objective function $L(W)$ is a collection from all the domain channels as follows:

$$\begin{aligned} L(W) &= \sum_d L_d(W_d) \\ &= \sum_d \log(P(d|x) \prod_{x,y} P(y|x,d)^{\overline{P}(x,y)}) \\ &= \sum_d \overline{P}(x,y) \log P(d|x) \\ &+ \sum_d \sum_{x,y} \overline{P}(x,y) \sum_k W_K F_k(x,y,d) \\ &- \sum_d \sum_x \overline{P}(x) \log \sum_y e^{\sum_k W_k F_k(x,y,d)} \end{aligned} \quad (4)$$

where $\overline{P}(x,y)$ is the empirical distribution that can be calculated from training samples. Hence, the gradient can be derived with feature vector $F(\cdot)$ as follows:

$$\begin{aligned} \frac{\partial L}{\partial W} &= \sum_d \sum_{x,y} \overline{P}(x,y) F(x,y,d) \\ &- \sum_d \sum_{x,y} \overline{P}(x) P(y|x,d) F(x,y,d) \end{aligned} \quad (5)$$

3.6 Model training

Since the joint representations learning in the proposed model is determined by both global domain distribution of input data X and the local context features, multi-task learning is applied for training with two different but related tasks: domain label classifier (a sub-task), and the sequence labeling model generation (main-task). For each batch in an epoch, the two tasks are trained alternatively. We firstly train the multi-class classifier using forward pass to calculate the loss $J(\theta)$ as eq.(3) and apply backward propagation to update the parameters θ. After the domain distribution is obtained, the joint representation s_i can be computed, and then the objective $L(W)$ of labeling model can be computed via domain-aware CRF as eq.(4). Finally the gradient can be computed as eq.(5) and back-propagation is deployed to update the parameters W and domain embedding v_i of main-task. Note that training will be stopped if the performance on the development set is no longer improving for 15 epochs or the maximum number of iterations has been achieved. The iterative training process is clarified in Algo.1.

4 EXPERIMENTS

4.1 Dataset

In order to validate the proposed method for sequence labeling, we use two types of datasets: one is blended data crossing multiple domains, and the other is a single data set without domain labels. The former ones are generated by combining several independent data sets following the works of Kurata et al. [13] and Zhai et al. [30]. Three considerably large data sets, LARGE3, LARGE4, and LARGE5 are obtained for the tasks of *named entity recognition (NER)* and *segmentation.* LARGEk represents how many sub sets it contains. For the latter one, we use CoNLL 2000 shared task for *text chunking* to test the effectiveness of our model on the data without specific domain information. For comparison, several baselines are selected

Table 1: Details of each sub set in LARGEs.

	training set	development set	test set	label scale	domain
ATIS	3,983	995	893	64	airline travel
MIT Restaurant	3,270	775	1521	9	restaurant
MIT Movie	7,726	2,049	2,443	13	movie
SemEval.2017.t10	2,418	415	847	4	scientific publications
SemEval.2018.t8	9,424	1,213	618	4	cyber-security reports

Algorithm 1 Training algorithm

```
randomize $v_i$,$\theta$,$W$
for each epoch do
    if sub-task then
        forward and backward pass based on the loss $J(\theta)$ as eq.(3)
    end if
    if main-task then
        for each domain $i$ do
            calculate$P(d_i|x) = softmax_i(f)$
            calculate $s^i$ as eq.(2)
        end for
        forward pass $L(W)$ as eq.(4)
        backward pass $\nabla_W L(W)$ as eq.(5)
    end if
    update all parameters
    stop if early stopping condition satisfied
end for
```

including classic linear probabilistic models and neural networks. The details of datasets and baselines are depicted as follows.

LARGEs. The construction of this kind of data set is aiming for containing a large amount of data sequences from several domains, which is motivated by the fact that many practical applications of sequence tagging involve a large scale of documents crossing multiple domains. Specifically, five open data sets for NER and segmentation are included: ATIS, MIT Restaurant Corpus, MIT Movie Corpus, SemEval 2017 task 10 and SemEval 2018 task 8. LARGEk denotes the combination of k-headed sub sets, and we collect the training set, development set and test set from each subset respectively for constructing the corresponding samples of LARGEs. The detail of each involved dataset are shown as Tab.1 and all the subsets can be found online.

CoNLL 2000. To verify our model can be applied to the data without domain labels, we use CoNLL 2000 to test the effectiveness. This data set is extracted from Wall Street Journal corpus, containing 8,068 training, 868 development and 893 test sentences. 12 syntactic labels are defined for text chunking, e.g. NP, VP, ADJP, other, etc.

4.2 Baselines

For comparison, several baselines are selected including classic linear probabilistic models and neural networks. We first choose CRF [14] as the classical linear probabilistic model baseline for its broad applications. Then, three recent state-of-art neural models, BiLSTM-CRF [11], LMcost [23], and LM-LSTM-CRF [21], are selected as the deep learning baselines. All of those models are implemented using open-source code offered by authors and we test them on the LARGEs and CoNLL 2000 data sets to evaluate the algorithms' performance.

4.3 Preparation

The proposed model applies same settings of pre-trained word embeddings and optimization strategy as baseline models referred before. To initialize the word lookup table, we deploy GloVe [20], 300-dimension pre-trained word embeddings, and the character/domain embeddings are both 100-dimension. To avoid bias, each test is repeated 10 times for averaging the final score. In LARGEs, each sample has a specific field label because its original data source covers a certain field as shown in Tab.1. However, unlike LARGEs, samples in CoNLL 2000 do not associate with explicit domain labels, hence, we apply K-means clustering to explore the distributional distinctiveness among the data sequences and labeled them with the cluster index. After the label system has been defined, the domain knowledge extractor is trained as a multi-class classifier. Then the labeling model can work based on the domain distribution.

The novel approach in this paper is noted as DA-BiLSTM-CRF++ in the following sections. In order to validate the usefulness of the domain knowledge in the sequence labeling tasks, we launch another two experiment models, DA-BiLSTM-CRF and DA-BiLSTM-CRF+, which simplify the joint representation of the proposed model. The key difference between these models and the DA-BiLSTM-CRF++ is whether the second layer of Bi-LSTM and the enhanced highway layer participate the joint representation learning. For details, joint representation extractor in DA-BiLSTM-CRF directly connects the contextual representation with domain embedding, and feeds the domain-aware CRF with the vector $[\overrightarrow{o}^1_t, \overleftarrow{o}^1_t, v_i]^T$. While the joint representation extractor in DA-BiLSTM-CRF+ applies the second layer of Bi-LSTM to learn a joint representation $[\overrightarrow{o}^2_t, \overleftarrow{o}^2_t]^T$ from the direct concatenation $[\overrightarrow{o}^1_t, \overleftarrow{o}^1_t, v_i]^T$ to feed the top domain-aware CRF layer. Moreover, DA-BiLSTM-CRF++ uses a multi-channel highway layer to balance the embeddings of the first and the second Bi-LSTM layer.

4.4 Experimental results

To evaluate the performance of the novel approach on the datasets with domain discrimination, we perform named entity recognition (NER) and segmentation on the test sets of LARGE3, LARGE4 and LARGE5. The performance is measured in terms of F1-score, which is calculated by the public script *conlleval.pl.* After the named entity

Table 2: The results of NER and segmentation on LARGEs. Note that * denotes p-value < 0.05 and ** denotes p-value < 0.005.

	NER			segmentation			chunking
	LARGE3	LARGE4	LARGE5	LARGE3	LARGE4	LARGE5	CoNLL 2000
CRF	80.42	76.54	70.27	85.19	76.54	73.89	86.18
BiLSTM-CRF	85.79	77.81	75.04	88.85	81.32	78.65	94.25
LMcost	86.08	77.65	75.39	89.47	82.67	79.87	94.03
LM-LSTM-CRF	86.2	77.98	75.48	89.67	83.22	79.98	94.41
DA-BiLSTM-CRF	86.71	78.12	75.47	89.68	82.54	79.9	94.7
DA-BiLSTM-CRF+	86.67	**78.6****	75.68	**89.91***	83.38	80.18	94.89
DA-BiLSTM-CRF++	**86.89****	78.53	**75.72***	89.83	**83.43***	**80.35****	**94.95***

labels have been obtained, we reassign each word with IOB labels to assess the segmentation performance. Also, we report the F1-score of segmentation which is computed by *conlleval.pl* in the Tab.2. The results indicate the novel model produces significant improvements (p-value < 0.05) due to the effects of the domain knowledge. DA-BiLSTM-CRF++ and DA-BiLSTM-CRF+ perform better than the basic version DA-BiLSTM-CRF in most of cases, because the former two models learn the joint embedding information via contextual and domain representation rather than a direct concatenation of them. DA-BiLSTM-CRF+ is functionally equivalent to DA-BiLSTM-CRF++ and performs roughly the same.

The robustness of the proposed model has also been tested. Several experiments on CoNLL 2000 shared dataset are deployed for testing the chunking performance, which is a typical sequence labeling task as well as NER or segmentation. The best F1-score is achieved with the parameter $k = 2$ when K-means clustering has been applied as a pre-processing to obtain the scattering distinctiveness of unlabeled data. In the same way, we report the F1-score via *conlleval.pl*, and the chunking results are displayed in Tab.2. The evaluation results show the effectiveness of proposed model (p-value < 0.05). When the explicit training labels are not available, the proposed approach of obtaining (pseudo) domain knowledge is promising. However, the performance is vulnerable to the incorrect clustering labels.

4.5 Analysis.

The model is validated for three main sequence labeling tasks by the experiments aforementioned, and the essential reasons of the superiority against prior works should be further investigated. As Fig.1 illustrated, the fundamental distinction of the proposed approach is the contribution of both the sequence features and the domain knowledge, which contains two important factors. First, we employ the joint representation of the input sequence X and domain knowledge d; second, the marginal probability calculated by domain-aware CRF based on eq.(1) is utilized to enhance the labeling performance. The improvements produced by the first point can be observed through the comparisons between the basic model DA-BiLSTM-CRF and DA-BiLSTM-CRF+/DA-BiLSTM-CRF++ that deploy some extra neural networks to learn the joint representation from the direct connection of contextual and domain embeddings. The second factor can be proved via the advantages of the basic model DA-BiLSTM-CRF in the experiments using the redesigned CRF layer to integrate each domain component. Furthermore, we investigate the correlation between performance and the class divergence in each dataset using Silhouette Coefficient [12], which is widely deployed as a relocating measurement. Tab.3 reveals the positive relationship between Silhouette Coefficient and the model performance, which proves that our method maximizes its potential if the experiment corpus can be clearly divided into a number of independent partitions (as domain knowledge). Note that S.C. represents the Silhouette Coefficient and F1-gap is the gap between the best-performed and the second-performed F1-scores.

Table 3: The relationship between Silhouette Coefficient and F1-gap on each dataset.

	S.C.	F1-gap
LARGE3	0.2128	0.69
LARGE4	0.2042	0.62
LARGE5	0.1246	0.24
CoNLL 2000@$k = 2$	0.2671	0.54
CoNLL 2000@$k = 3$	0.1249	0.49
CoNLL 2000@$k = 4$	0.0807	0.31
CoNLL 2000@$k = 5$	0.0610	0.17

5 CONCLUSION AND FUTURE WORK

In this study, we propose a novel neural sequence labeling model in a joint learning framework. While the model takes the domain probability distribution into account, the global and local features are integrated via a novel multi-channel highway learning for efficient joint embedding optimization. Moreover, an innovative CRF layer integrates the joint representations that range over all the possible domains and calculates the probability of the output labels. The proposed method is trained in a multi-task learning framework. Three datasets (with or without explicit domain labels) associated with different sequence labeling tasks were employed for algorithm evaluation. Experiment result indicated that the proposed model significantly outperforms the previous classical and recent baselines in a number of different sequence labeling tasks, like NER, segmentation and chunking. In this work, clustering based approach was implemented to address the problem of domain label missing, and in the future, we will investigate more sophisticated end-to-end method to consider domain knowledge into sequence labeling task when explicit domain labels are not available.

REFERENCE

[1] Kai-Wei Chang, Scott Wen-tau Yih, Bishan Yang, and Chris Meek. 2014. Typed Tensor Decomposition of Knowledge Bases for Relation Extraction. (2014).

[2] Jason PC Chiu and Eric Nichols. 2015. Named Entity Recognition with Bidirectional LSTM-CNNs. *arXiv preprint arXiv:1511.08308* (2015).

[3] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *arXiv preprint arXiv:1409.1259* (2014).

[4] Arman Cohan and Nazli Goharian. 2017. Contextualizing Citations for Scientific Summarization Using Word Embeddings and Domain knowledge. *arXiv preprint arXiv:1705.08063* (2017).

[5] Michael Collins. 2013. The Forward-Backward Algorithm. (2013).

[6] 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 (2011), 2493–2537.

[7] Christoph Goller and Andreas Kuchler. 1996. Learning Task-Dependent Distributed Representations by Backpropagation Through Structure. In *Neural Networks, 1996., IEEE International Conference on*, Vol. 1. IEEE, 347–352.

[8] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech Recognition with Deep Recurrent Neural Networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 6645–6649.

[9] John M Hammersley and Peter Clifford. 1971. Markov fields on Finite Graphs and Lattices. (1971).

[10] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[11] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. *arXiv preprint arXiv:1508.01991* (2015).

[12] L. Kaufman and P.J. Rousseeuw. 1990. Finding Groups in Data: an Introduction to Cluster Analysis.

[13] Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu. 2016. Leveraging Sentence-Level Information with encoder LSTM for Semantic Slot Filling. *arXiv preprint arXiv:1601.01530* (2016).

[14] John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*. 282–289.

[15] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. *arXiv preprint arXiv:1603.01360* (2016).

[16] Liyuan Liu, Jingbo Shang, Frank Xu, Xiang Ren, Huan Gui, Jian Peng, and Jiawei Han. 2017. Empower Sequence Labeling with Task-Aware Neural Language Model. *arXiv preprint arXiv:1709.04109* (2017).

[17] Xuezhe Ma and Eduard Hovy. 2016. End-to-End Sequence Labeling via Bi-Directional LSTM-CNNs-CRF. *arXiv preprint arXiv:1603.01354* (2016).

[18] David JC MacKay. 1996. Equivalence of Linear Boltzmann Chains and Hidden Markov Models. *Neural Computation* 8, 1 (1996), 178–181.

[19] Andrew McCallum, Dayne Freitag, and Fernando CN Pereira. 2000. Maximum Entropy Markov Models for Information Extraction and Segmentation. In *ICML*, Vol. 17. 591–598.

[20] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543.

[21] Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-Supervised Sequence Tagging with Bidirectional Language Models. *arXiv preprint arXiv:1705.00108* (2017).

[22] Lawrence R Rabiner. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. IEEE* 77, 2 (1989), 257–286.

[23] Marek Rei. 2017. Semi-Supervised Multitask Learning for Sequence Labeling. *arXiv preprint arXiv:1704.07156* (2017).

[24] Lawrence K Saul and Michael I Jordan. 1995. Boltzmann Chains and Hidden Markov Models. In *Advances in Neural Information Processing Systems*. 435–442.

[25] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Highway Networks. *arXiv preprint arXiv:1505.00387* (2015).

[26] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in neural information processing systems*. 3104–3112.

[27] Howard Turtle and W Bruce Croft. 2017. Inference Networks for Document Retrieval. In *ACM SIGIR Forum*, Vol. 51. ACM, 124–147.

[28] Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit Semantic Ranking for Academic Search via Knowledge Graph Embedding. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1271–1279.

[29] Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-Task Cross-Lingual Sequence Tagging from Scratch. *arXiv preprint arXiv:1603.06270* (2016).

[30] Feifei Zhai, Saloni Potdar, Bing Xiang, and Bowen Zhou. 2017. Neural Models for Sequence Chunking.. In *AAAI*. 3365–3371.