BIDIRECTIONAL LONG SHORT-TERM MEMORY (BILSTM)WITH CONDITIONAL RANDOM FIELDS (CRF) FOR KNOWLEDGE NAMED ENTITY RECOGNITION IN ONLINE JUDGES (OJS)

[1]Muhammad Asif Khan, [2, 3]Tayyab Naveed, [1]Elmaam Yagoub, [1]Guojin Zhu

[1]School of Computer Science and Technology, Donghua University Shanghai, China
[2]College of Textile Engineering, Donghua University, Shanghai, China
[3]School of Fine Art, Design & Architecture, GIFT University, Gujranwala, Pakistan

ABSTRACT

This study investigates the effectiveness of Knowledge Named Entity Recognition in Online Judges (OJs). OJs are lacking in the classification of topics and limited to the IDs only. Therefore a lot of time is consumed in finding programming problems more specifically in knowledge entities.A Bidirectional Long Short-Term Memory (BiLSTM) with Conditional Random Fields (CRF) model is applied for the recognition of knowledge named entities existing in the solution reports.For the test run, more than 2000 solution reports are crawled from the Online Judges and processed for the model output. The stability of the model is also assessed with the higher F1 value. The results obtained through the proposed BiLSTM-CRF model are more effectual (F1: 98.96%) and efficient in lead-time.

Keywords

Knowledge Named Entity Recognition; Online Judge; Machine Learning; BiLSTM-CRF

1. INTRODUCTION

In this modern era, online education has become renownedglobally. Moreover, due to the fast living style of humans, the time duration for the completion of activities is also a critical detrimental. The advent of machine learning has further enhanced the configuration of technical work projects. However, due to the non-existence of algorithmic knowledge entities, experts and beginners have difficulties while solving the programming problems of their interestin online judges. The Online Judges (OJs)have only described the IDs and titles of the problems. Theyhave not explicitly definedthe algorithmic knowledge that leadsto proper and easy solutions.Furthermore,a lot of time and resources of the users are wasted due to the nonexistence of knowledge named entities.

Online Judges (OJs) are the systems intended for reliable evaluation of algorithm source code submitted by the programmers.OJs execute and test the code, submitted in a homogenous environment and provide real-time assessment of the solution code submitted by the users.While Named Entity Recognition (NER) is a text tagging technique that automatically recognizes and classifiesthe words or phrases which described a prime concept (entities) in a sentence [1].NER assigns a label (class) or semantic category from a predefined set to the expression known as entity mention to describe the concept[2].

NER systems are divided into two groupsby Natural Language Processing (NLP) researchers. Group one is based on regular expression or dictionaries calledrule-based system[1]. They are expensive and unfeasible. Group two uses machine learning techniques and are more feasible and

DOI: 10.5121/ijnlc.2018.7401

fast. In recent years, a variety of machine learning approaches use deep neural network models have proposed and applied linguistic sequence tagging task,for example, POS tagging, chunking and NER[3],[4],[5]. While most of the statistical learning approaches such as SVM, CRF and perceptron models largely depend on feature engineering. Collobert et al. [5] presented SENNA, which employs a feed-forward neural network (FFNN) and word embeddings to accomplish near state of the art results. But the model proposed by them is a simple FFNN and only consider a fix size window that has miss long-distance dependency between words of a sentence. Lample et al. [4] and Chiu et al. [6] implemented BiLSTM model with CNN for NER. However, CNN was found unfeasible for long-term dependency in text instead of Long Short Term Memories (LSTM) for NER task. Similarly,Santos et al. presented neural architecture CNN to model character level information "CharWNN" structure [7]. RecentlyDernoncourt et.al.[8] have proposed a convenient named-entity recognition system based on ANNs known as NeuroNER.

In the literature readings, most of the frameworks for entity recognition arerule-basedand cost ineffective. Although the rule-basedapproaches are simple but high time to consume and difficult in a changeof domain are its major drawbacks. Further rule-based methods such as ANNIE [9] or SystemT [10]also required the manual development of grammar rules in order to identify the named entities. Therefore the trend isconverted from the manual rules based architectures to automatic machine learning techniques. The machine learning-based algorithms apply various approachesfor entity recognition. However these algorithm may require prior training data i.e. supervised learning for example decision trees [11], support vector machines (SVM) [12], conditional random fields (CRFs) [13], hidden Markov model (HMM) [14] etc., or they may be totally unsupervised and needs no training data [15]. These machine learning techniques are highly efficient and outperform rule-based approaches.

In our neural network,CRF is used with BiLSTM model.The problem of dependency on words is controlled by using bidirectional LSTM, which has both forward and backward propagation layers. More than 2000 solution reports were extracted from two Online Judges and processed for the neural network model. The model has input (text files) and predicts tag for each word in the sentence.The strength of the proposed modelis characterized by the learning ability of the model.

2.1 RESEARCH METHODOLOGY

2.1 LSTM NETWORK

Figure 1has shown the Long short-term memory (LSTMs) network which is a modified form of Recurrent Neural Network [16]. The traditional RNNs are sensitive to the gradient. Therefore LSTM was usedwhich has prolonged memories(due to gates and one cell state) and capable of processing variable-length input vector. Furthermore,BidirectionalLSTM networks can better deal with contextual dependency.BiLSTMhas two propagating networks in opposite direction, one network runs from the beginning of the sentence to the end while the other network works in the backward direction.These forward and backward networks memorize the information about the sentence from both directions. The BiLSTM uses one layer for forward and the other for backward LSTM. The dimensions of the BiLSTM were set to 100.

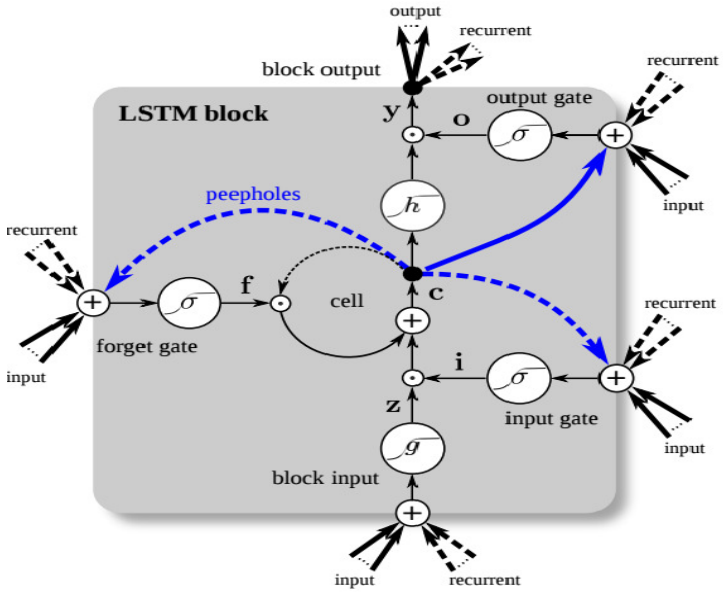

Figure 1 The basic LSTM structure

The prime idea of using BiLSTM is to extract a feature from a sentence by capturing information from both previous and next states respectively and merge the two hidden states to produce the final output.

$$h_i = [h_{i-1};\ h_{i+1}]$$

2.2 CRF

The CRFs are used in NLP problem to conduct the segmentation and labeling of data. They were first introduced in 2001 by Lafferty et al. [13].The CRF layer take the hidden states $h = (h_1, h_2, \dots, h_n)$ as input from the BiLSTM layer, and produce the tagged output as final prediction sequence. CRFs focused on sentence level (i.e. considering both the previous and next words) instead of an individual position in predicting the current tag and thus have better accuracy.

2.3 BILSTM-CRF MODEL

Figure 2has shown the proposed BiLSTM-CRF model for Knowledge Named Entity Recognition.First solution reports are extracted from OJs in HTML form. Thesesolution reports are further processed and converted to text format. The input to the BiLSTM layer are vector representations of each single word x_n,i.e. $(x_1, x_2, \dots, x_n)$. The vectors are fed to the BiLSTM network at each time step *t*. Next, the output vectors of BiLSTM layer h_i which are formed by the concatenation of forward hidden state $\vec{h}$ and backward hidden state $\overleftarrow{h}$ are fed to the CRF layer to jointly decode the best label sequence for each word.This network efficiently use past and future input features via a BiLSTM layer and sentence level tag information through a CRF layer.

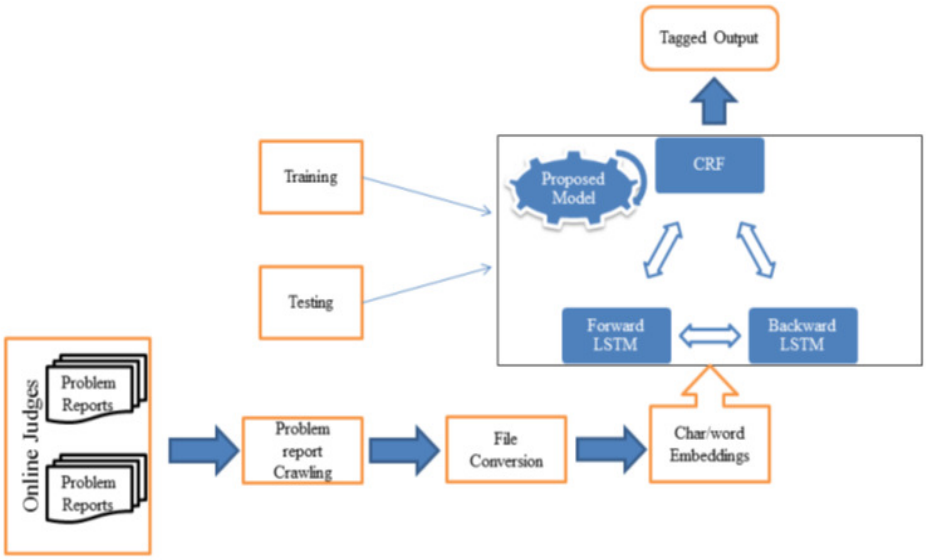

Figure 2 Proposed BiLSTM-CRF model for Knowledge Named Entity Recognition

It is more beneficial to apply word embeddings than arbitrary initialized embeddings. The reason is thatthe pre-trained word embeddings largely affects the execution of the model, which has significantly bigger effect than numerous different hyperparameters. In our experiment, the available Global Vector (GloVe) word embeddings was used to initialize the lookup table. The GloVe word representations are trained on 6 billion words from Wikipedia and Web text [17].

2.4 LEARNING METHOD

Training was performed with stochastic gradient descent (SGD) with different learning rates. The initial learning rate was kept at 0.005. Further, this learning rate was updated at each training epoch. The applied optimization algorithmhas better results as compared with Adadelta [18] and RMSProp [19]. To avoid the over-fitting (low accuracy) of the system, dropout method was applied [20]. Dropout is a method used in a neural network where neurons are randomly selected to be ignored during training. They are "dropped-out" randomly which is very helpful in NER tasks as reported in [20] and [4].In our experiment, the dropout value was kept to 0.5. Table 1 has shown the various hyper-parameters adjustments for our experiment. The BiLSTM-CRF model was implemented through Tensorflow (machine learning library in Python).The experiments were conducted on GEFORCE GTX 1080 Ti GPU with Ubuntu 16.04 LTS. The proposed model required 3 to 4 hours for training and testing. Table 1 has shown the parameters used in the model.

Table 1 Hyper-parameters values used in the experiment

Hyper-parameters	Value
Initial state	0
Learning rate	0.005-0.05
Drop out	0.5
Character embedding dimension	25
Word embedding dimension	100
Gradient clipping value	5.0

2.5 TAGGING SCHEME

The BIOES tagging scheme was applied since it is the most expressive scheme for labeling the words in a sentence. Similarly to part of speech tagging here for knowledge named entity recognition aims to assign a label to every word in the sentence.

3. RESULTS AND DISCUSSION

3.1 DATASET

For the data set, more than 2000 reports were extracted from two online judges (OJs) by using web crawler. These crawled problem solution reports have contained hundreds of knowledge entities belonging to different classes of knowledge named entities. Additionally, the crawled HTML documents were preprocessed and various inconsistencies have been removed. The HTML documents were converted into text files which were used as input for the proposed model. The data was divided into three sets i.e. training, testing and validation set. Table 2 has given the distribution of the whole corpus into three parts. Moreover,the knowledge named entities was subdivided into seven classes. Each knowledge named entity in the document was tagged with any of the six types; MATH, SORTING, TREE, MAP, DP or SEARCH and labeled as OTHER if doesn't belong to anyone of them.

Table 2 Training testing and validation set

Set	No. of Tokens
Train	309998
Test	91384
Validate	72152

CoNLL-2003 standard metrics i.e. precision, recall,and F1, were applied for the evaluation of proposed BiLSTM-CRF model. Precision, recall, and F1-score wereestimated through:

$$\text{precesion} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (1)$$

$$recall = \frac{TP}{TP + FN} \quad (2)$$

$$F1 - score = \frac{2 * precesion * recall}{precesion + recall} \quad (3)$$

Where TP: the number of true positives, FP: the number of false positives, and FN: the number of false negatives.

Table 3 has presented the comparison of BiLSTM-CRF model with other models for knowledge named entity recognition. The precision (98.25%), recall (96.57%) and F1 (98.96%) have been achieved on test data set. The outcomes of the experiment were compared with the best results of Zhu et al. [21], which were obtained by building a knowledge base and deploying CNNs. It was observed that the precision was lower by 0.22%, however, recall and F1 has higher valuesthan the previous published best performance results. There was an increase of 0.30% in recall and 0.85%

in F1 value. Additionally, BiLSTM-CRF model was faster and have less feature dependence as compared to the earlier model. Our model also obtained competitive performance as compared to [22] and [23] where they applied word embeddings and BiLSTM network.

Table 3 Comparison with the previous best model on the test set

Model	Word Embeddings	Precision	Recall	F1	Time Taken
CNN	Word2vec	98.47	96.27	97.36%	20 hrs
BiLSTM-CRF	Glove	98.25	96.57	98.96%	6 hrs

The data was processed several times to acquire more efficient results by tuning the parameters. Thus theneural network model was trained and tested with a variety of different hyperparameters in order to get better and robust knowledge named entity recognition systems. However, it's a time-consumingprocess.Since the training and tuning of neural networks sometimes takemany hours or even days.Figure 3 has shown the F1 score versus the number of epochs of the BiLSTM-CRF network.

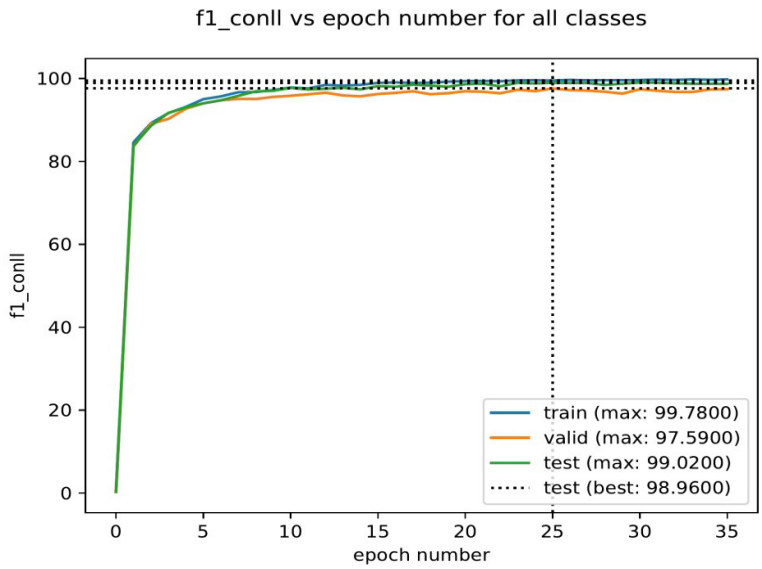

Figure3 F1 vs. epoch number on the train, test and validate set

Figure 3 shows that in the first few epochs the model learns very quickly and the F1 value approached 80%. The later epochs are shown very little improvements. The number of epochs was set to 40, while the learning algorithm i.e. stochastic gradient descent started to overfit at 35th epoch. Therefore *early stopping* method was applied for regularizing the model. In other words, early stopping guides to avoid the learner from overfitting. Further, it was noted that the performance of the model was sensitive to learning rate. The neural network converged quickly if the learning rate was higher. It was also observed that the model leads to early stopping at epoch 13 while the learning rate was kept to 0.05. The reason for early stopping was due to the sensitivity of stochastic gradient descent (SGD) optimizer towards the learning rate.

4. CONCLUSION

In this research work, the proposed Bidirectional Long Short-Term Memory (BiLSTM) with Conditional Random Fields (CRF) model wassuccessfully applied for the recognition of knowledge named entities existing in the solution reports.It is an important contribution to the field since the BiLSTM-CRF architecture hasbetter performance i.e. F1 (98.96%) and has no dependency on hand-crafted features.Furthermore, the model suggest that BiLSTM-CRF is a better choice for sequence tagging and was also found efficient atruntime.

REFERENCES

[1] Nadeau, D. and S. Sekine, A survey of named entity recognition and classification. Lingvisticae Investigationes, 2007. 30(1): p. 3-26.

[2] Piskorski, J. and R. Yangarber, Information extraction: past, present and future, in Multi-source, multilingual information extraction and summarization. 2013, Springer. p. 23-49.

[3] Lopez, M.M. and J. Kalita, Deep Learning applied to NLP. arXiv preprint arXiv:1703.03091, 2017.

[4] Lample, G., et al., Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360, 2016.

[5] Collobert, R., et al., Natural language processing (almost) from scratch. Journal of Machine Learning Research, 2011. 12(Aug): p. 2493-2537.

[6] Chiu, J.P. and E. Nichols, Named entity recognition with bidirectional LSTM-CNNs. arXiv preprint arXiv:1511.08308, 2015.

[7] Santos, C.D. and B. Zadrozny. Learning character-level representations for part-of-speech tagging. in Proceedings of the 31st International Conference on Machine Learning (ICML-14). 2014.

[8] Dernoncourt, F., J.Y. Lee, and P. Szolovits, NeuroNER: an easy-to-use program for named-entity recognition based on neural networks. arXiv preprint arXiv:1705.05487, 2017.

[9] Cunningham, H., et al. A framework and graphical development environment for robust NLP tools and applications. in ACL. 2002.

[10] Chiticariu, L., et al. SystemT: an algebraic approach to declarative information extraction. in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. 2010. Association for Computational Linguistics.

[11] Quinlan, J.R., Induction of decision trees. Machine learning, 1986. 1(1): p. 81-106.

[12] Joachims, T. Text categorization with support vector machines: Learning with many relevant features. in European conference on machine learning. 1998. Springer.

[13] Lafferty, J., A. McCallum, and F.C. Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data. 2001.

[14] Zhou, G. and J. Su. Named entity recognition using an HMM-based chunk tagger. in proceedings of the 40th Annual Meeting on Association for Computational Linguistics. 2002. Association for Computational Linguistics.

[15] Nadeau, D., P.D. Turney, and S. Matwin. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. in Conference of the Canadian Society for Computational Studies of Intelligence. 2006. Springer.

[16] Hochreiter, S. and J. Schmidhuber, Long short-term memory. Neural computation, 1997. 9(8): p. 1735-1780.

[17] Pennington, J., R. Socher, and C. Manning. Glove: Global vectors for word representation. in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.

[18] Zeiler, M.D., ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701, 2012.

[19] Hinton, G., N. Srivastava, and K. Swersky, Rmsprop: Divide the gradient by a running average of its recent magnitude. Neural networks for machine learning, Coursera lecture 6e, 2012.

[20] Srivastava, N., et al., Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 2014. 15(1): p. 1929-1958.

[21] Zhu, G. and P. Shen, Identification discovery of algorithmic knowledge entity based on deep learning. Intelligent Computer & Applications, 2017.

[22] Yao, Y. and Z. Huang. Bi-directional LSTM recurrent neural network for Chinese word segmentation. in International Conference on Neural Information Processing. 2016. Springer.

[23] Huang, Z., W. Xu, and K. Yu, Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991, 2015.