IEICE TRANS. INF. & SYST., VOL.E101–D, NO.3 MARCH 2018

PAPER

Corpus Expansion for Neural CWS on Microblog-Oriented Data with λ-Active Learning Approach

Jing ZHANG[†a)], **Degen HUANG**[†b)], **Kaiyu HUANG**[†c)], **Zhuang LIU**[†d)], ***Nonmembers***, ***and*** **Fuji REN**[††e)], ***Fellow***

SUMMARY Microblog data contains rich information of real-world events with great commercial values, so microblog-oriented natural language processing (NLP) tasks have grabbed considerable attention of researchers. However, the performance of microblog-oriented Chinese Word Segmentation (CWS) based on deep neural networks (DNNs) is still not satisfying. One critical reason is that the existing microblog-oriented training corpus is inadequate to train effective weight matrices for DNNs. In this paper, we propose a novel active learning method to extend the scale of the training corpus for DNNs. However, due to a large amount of partially overlapped sentences in the microblogs, it is difficult to select samples with high annotation values from raw microblogs during the active learning procedure. To select samples with higher annotation values, parameter λ is introduced to control the number of repeatedly selected samples. Meanwhile, various strategies are adopted to measure the overall annotation values of a sample during the active learning procedure. Experiments on the benchmark datasets of NLPCC 2015 show that our λ-active learning method outperforms the baseline system and the state-of-the-art method. Besides, the results also demonstrate that the performances of the DNNs trained on the extended corpus are significantly improved.

key words: *Chinese word segmentation, active learning, deep neural networks, corpus expansion*

1. Introduction

Chinese Word Segmentation (CWS) is a prerequisite task in Chinese natural language processing (CNLP). The task was treated as a sequence labeling problem and solved using the Maximum Entropy Markov Model (MEMM) by Xue et al. [1]. Later, other conventional sequence labeling models, such as Conditional Random Fields (CRFs) [2] and Hidden Markov Model (HMM), have been applied to the CWS task and obtained outstanding performances. Recently, deep neural networks (DNNs) have attracted increasing attention in natural language processing (NLP) fields for their strength in minimizing the efforts in feature engineering. DNNs have been widely used in CWS tasks [3]–[5], after Collbert et al. [6] proposed a neural network architecture outperforming the state-of-the-art systems on a variety of sequence labeling problems.

Both conventional sequence labeling models and DNNs for CWS tasks have achieved a great progress on traditional newswires datasets, owing to the large scale shared manual corpora. However, the performance of CWS approaches is still not satisfying on informal text, for example microblog-oriented data [7]–[10].

Recently, microblog-oriented NLP tasks have grabbed considerable attention of researchers, such as microblog-oriented sentiment analysis [11], named entity recognition from microblogs [12]–[14], microblog retrieval [15]. CWS is the prerequisite step of microblog-oriented NLP tasks. To promote the research in microblog-oriented CWS, the shared tasks of microblog-oriented CWS are added into NLP conferences, such as NLPCC and COAE. Among those CWS approaches submitted to solve the shared tasks, DNNs such as recurrent neural networks (RNNs) and long short-term memory (LSTM) neural networks are used, but the performances of these neural CWS approaches are not significantly better, rather even worse than conventional models. The critical reason for this phenomenon is the scale of the existing microblog-oriented training corpus is inadequate for training the parameter matrices of DNNs. The purpose of this paper is to utilize the active learning approaches to select samples with high annotation values from raw microblog datasets to extend the training corpus for neural CWS.

Active learning approaches have been used in many NLP fields to extend training corpora [16]–[18]. To date, the active learning approaches have obtained prominent results on selecting samples for CWS tasks on traditional newswires datasets. But it is difficult to effectively select samples from unlabelled microblogs, because the scale of unlabelled microblogs are too large and the quality of samples in microblog datasets is various. For instance, there are many samples which are partially overlapped but not identical (shown as **Sample A** and **Sample B**) in microblog datasets.

Sample A: #微评#弱势群体怎能被 “弱视”? (#Micro comment# How can the vulnerable groups be “neglected”?)

Sample B: #微评#拍饭有风险! (#Micro comment# It is risky to dine out with strangers!)

During the sample selecting procedure of the active learning approach, it is difficult to measure the diversity for the partially overlapped samples using the traditional active

Manuscript received July 21, 2017.
Manuscript revised November 8, 2017.
Manuscript publicized December 8, 2017.

[†]The authors are with the School of Computer Science and Technology, Dalian University of Technology, Dalian, Liaoning 116024, China.
[††]The author is with the Department of Information Science & Intelligent Systems, Tokushima University, Tokushima-shi, 770–8506 Japan.
a) E-mail: zhangjingqf@mail.dlut.edu.cn
b) E-mail: huangdg@dlut.edu.cn (Corresponding author)
c) E-mail: huangkaiyucs@foxmail.com
d) E-mail: zhuangliu@mail.dlut.edu.cn
e) E-mail: ren@is.tokushima-u.ac.jp
DOI: 10.1587/transinf.2017EDP7239

learning approach: Assuming that **Sample A** has already been selected and the "微" (Macro) or "评" (comment) in **Sample A** is the current observed object for evaluating other samples, when **Sample B** comes, the contexts of the current observed object in **Sample A** are exactly the same with the contexts in **Sample B**, which is red in the samples. Therefore, the diversity of **Sample B** is evaluated as low. As a result, **Sample B**, which contains a new word "拍饭" (dine out with strangers), will be filtered out by the traditional active learning approach.

Aiming to efficiently select samples with high annotation values from raw microblogs, we propose a novel active learning approach, λ-active learning approach. We introduce the parameter λ to control the number of the selected partially overlapped samples and adopt various strategies to evaluate the overall annotation values of a sample during the sample selecting procedure. Besides, we choose the CRFs model as the initial segmenter in the active learning procedure to select samples for DNNs to avoid the efforts for parameter tuning and training. The experimental results show that our λ-active learning method outperforms the-state-of-art method, and the training corpus obtained by our method can significantly improve the segmentation performance of neural CWS on microblog corpus.

2. Related Work

2.1 Neural CWS Tasks

Recently, neural CWS have attracted increasing attention. Pei et al. [4] proposed a Max-Margin Tensor Neural Network (MMTNN) for CWS tasks, which can model complicated interactions between tags and context characters and speed up the model and avoid over-fitting. Chen et al. [19] proposed a Gated Recursive Neural Network (GRNN) segmentation model, incorporating the complicated combinations of context characters by reset and update gates. In order to gain long-distance information, various long short-term memory (LSTM) neural networks were proposed to get local and long-distance dependency information of current observed tokens, and the experimental results showed that the LSTM neural networks outperform other DNNs [20]–[22]. Therefore, we employ the LSTM layers instead of other DNNs in our experiments.

2.2 Active Learning for CWS Tasks

Active learning approaches have already been widely used in corpus expansion tasks. Li et al. [23] introduced the Word Boundary Annotation (WBA) method to evaluate the uncertainty confidence of character labels based on the edge probability of the CRFs model. According to the WBA method, if the sequence label is the right boundary of a word, it will be denoted as Y containing E (end) and S (single); otherwise, it will be denoted as N containing B (begin) and M (middle). After that, the post-probabilities of these two categories are calculated by adding the edge probabilities of the labels in the corresponding category. As a result, the uncertainty confidence of the observed character label is computed according to Eq. (1):

$$H(c) = \max_{x \in \{N,Y\}} \{P_x(c) - 0.5\} \tag{1}$$

where c is the current character; $P_x(c)$ denotes the post-probability of character c being annotated as category x. The lower the uncertainty confidence value, $H(c)$, is, the more informative the boundary is, and the higher the annotation value of this character is.

Liang et al. [24] proposed an active learning method based on Nearest Neighbor (ALN), which constructs nearest neighbor sets by calculating average Euclidean distance between samples and selects the samples according to Information Entropy (IE). Feng et al. [25] proposed an active learning segmentation algorithm to select samples from those with confidence higher than the threshold using a pool-based strategy.

The above active learning approaches have achieved outstanding results on selecting samples from traditional datasets for CWS tasks. But the performance is not satisfying on microblog datasets due to the sample characteristics of microblogs. In order to effectively select samples from raw microblogs for neural CWS, we propose the λ-active learning method to measure the context diversity of the character and utilize three strategies to evaluate the overall annotation value of a sample.

3. LSTM-Based CWS Architecture

As LSTM can capture long-distance context information, we employ the LSTM layers in our experiments. In this section the LSTM-based CWS architecture is introduced and is shown in Fig. 1.

The input of the architecture is the context of the current character in the observed sequence (the input sentence),

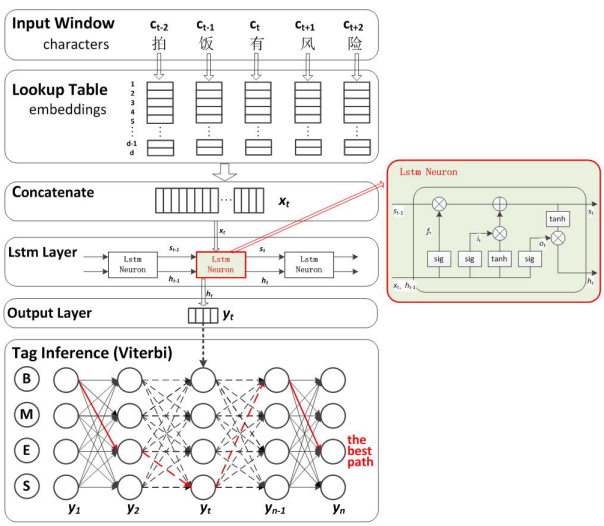

Fig. 1 LSTM-based CWS architecture

Table 1 Settings for training the LSTM neural networks

Hyperparameters	Settings value
Mini-batch size	20
window size	5
the number of hidden units	150
character embedding size	100
dropout rate	0.2

and the size of the context is pre-set, which is 5 in our experiments. The embeddings of characters in the context, which are randomly generated in the training procedure, are concatenated to feed the LSTM layer.

In the LSTM layer, the neuron is controlled by three gates: input gate i_t, forget gate f_t, and output gate o_t. The inputs of the LSTM neuron consist of x_t, s_{t-1}, and h_{t-1}, where x_t represents the concatenated character embedding, s_{t-1} represents the state value of the last neuron, and h_{t-1} represents the output value of the last neuron. Their calculating formulas are as follows:

$$i_t = \sigma(W^i x_t + U^i h_{t-1} + V^i s_{t-1} + b^i) \quad (2)$$

$$f_t = \sigma(W^f x_t + U^f h_{t-1} + V^f s_{t-1} + b^f) \quad (3)$$

$$o_t = \sigma(W^o x_t + U^o h_{t-1} + V^o s_{t-1} + b^o) \quad (4)$$

$$s_t = f_t \odot s_{t-1} + i_t \odot tanh(W^u x_t + U^u h_{t-1} + b^u) \quad (5)$$

$$h_t = o_t \odot tanh(s_t) \quad (6)$$

where i_t, f_t, and o_t represent the gate vector of input gate, forget gate, and output gate respectively; σ represents sigmoid function, $\odot$ represents element-wise multiplication; W^i, U^i, V^i, W^f, U^f, V^f, W^o, U^o, V^o, W^u, and U^u represent the weight matrices of corresponding gates. b^i, b^f, b^o, and b^u represent the biases of corresponding gates.

The output of the LSTM layers is the edge probability of the current observed character being tagged as each label in the sequence label set (B, M, E, S). In the process of tag inference, as usual, the output of the LSTM neural networks is predicted directly by choosing the label with biggest edge probability for the current observed character without taking into account the relationship and restrictions of the context's labels (for instance, B should not appears after M, and S should not appears after B). In our LSTM-based CWS architecture, we employ Viterbi algorithm to address this problem.

Since we are interested in the general influence of the extended training corpus on LSTM neural networks, the hyper-parameters of the LSTM neural networks were set as per previous works [5], [20] instead of being fine-tuned. The settings for training the LSTM neural networks are shown in Table 1. Besides, in the process of training the LSTM neural networks, we randomly extract 10% sentences from the original training set as the validation set, and the remained 90% sentences are used as the training set.

4. λ-Active Learning Algorithm

4.1 Semi-Supervised Initial Segmenter

An initial segmenter is required and needs to be retrained many times in the entire active learning procedure. Therefore it's better to choose a segmenter which is not very time-consuming as the initial segmenter. Considering that with the same training set, one learning epoch of LSTM neural networks takes about 3 hours, while that of CRFs model takes just a couple of minutes, we finally use the CRFs model as the initial segmenter. To train the initial segmenter, the context characters of the current observed character are utilized as comment features. Since semi-supervised machine learning methods can significantly improve the performance of the CRFs-based segmenter [26], [27], semi-supervised features are also adopted in this paper to take advantage of information extracted from the large scale unlabelled corpus.

4.1.1 Point-Wise Mutual Information (PMI)

In CWS tasks, PMI is used to measure the relatedness of two adjacent characters in the unlabelled corpus, and its calculating formula is shown as Eq. (7):

$$PMI(x, y) = \log \frac{P(x, y)}{P(x)P(y)} \quad (7)$$

where x and y represent the characters in the corpus; $P(x, y)$ denotes that the probability of character x appearing together with y as adjacent strings; $P(x)$ and $P(y)$ are the probability of x and y appearing in the whole corpus, respectively.

With Eq. (7), we can calculate the PMI value for all adjacent characters in the unlabelled corpus. Since the features that feed into CRFs models should be discrete, we round up the PMI values as the features. For instance, $(C_0, \lceil PMI(C_{-1}, C_0) \rceil)$ and $(C_0, \lceil PMI(C_0, C_1) \rceil)$ are used to feed CRFs models as the PMI features of current observed characters C_0.

4.1.2 Anti-Word Probability (AP)

For training the CRFs model, we also introduce another statistic, AP, to measure the possibility of a character being an anti-word. In order to calculate the AP value of the current observed character, character embeddings are pre-trained by using the unsupervised word2vec model. To train character embeddings, 300 thousand unlabeled microblogs are collected and segmented by characters. In our experiment, the training parameters of word2vec model are: dimension =200, window =9, minimum word frequency =1.

To calculate the AP value of the current character, we construct an anti-word set by utilizing the pre-trained character embeddings and a small anti-word seed set. In the seed set there are 11 elements "我", "是", "的", "了", "在", "。", "，", "、", "；", "!", "?", which are commonly used as a single word and have high probability of being an anti-word. The construction of the anti-word set is specifically described as Algorithm 1.

The formula we proposed to calculate the AP value for the token is shown as follows:

Algorithm 1 Anti-word set construction algorithm

Input: character embeddings dictionary $WEDictionary$, anti-word seed set $AWSet$, corpus $segCorpus$.
Output: anti-word set.
for iterator = 1 to T **do**
  Initialize a list to save the character embeddings for the characters in $AWSet$: $SeedSetCE$ = [ ]
  for chara in $AWset$ **do**
    Get the character embeddings for the chara and put the embeddings into the list $SeedSetCE$.
  end for
  Initialize a dictionary to store the token and its AP value: $tokenAP$={}
  for Token in $segCorpus$ **do**
    if Token in $tokenAP$ **then**
      continue
    else
      Lookup embeddings for Token, denoted as $CE(T)$.
      Calculate the AP value for Token using $CE(T)$ and the embeddings in $SeedSetCE$ according to Eq. (8).
      Add AP value into the KEY (=Token) in $tokenAP$.
    end if
  end for
  Select top N_{AP} tokens from $tokenAP$ which is sorted by the tokens' AP value from high to low.
  Put the selected tokens into $AWSet$.
end for

$$AP(token, AWset) = \frac{1}{N}\sum_{i=1}^{N} sim(token, chara_i) \tag{8}$$

where N is the total number of tokens in anti-word seed set $AWset$; $chara_i$ is the i^{th} token in $AWset$;$sim(token, chara) = \frac{CE(token)\cdot CE(chara)}{|CE(token)||CE(chara)|}$, $CE(c)$ is the character embedding of token c, $|vector|$is the modulus of the vector.

According to Algorithm 1, the final anti-word set is obtained after T iterations (T=3, in our experiments), where there are about 200 tokens. After that, the anti-word set is used to create the AP features for CRFs models by calculating the AP value of the current observed token according to Eq. (8). The AP value is also discretized for feeding the CRFs model in accordance with the following scheme.

$$APCRFs = \begin{cases} -1, AP < 0 \\ -2, AP = 0 \\ \lceil AP * 10 \rceil, AP > 0 \end{cases} \tag{9}$$

4.2 Corpus Expansion with λ-Active Learning

Together with the initial segmenter, the uncertainty confidence and the context diversity of the character are of great importance in the active learning procedure as well.

4.2.1 The Uncertainty Confidence

To evaluate the uncertainty confidence of the character, we modified the WBA method proposed by Li et al. [23]. After dividing the CRFs annotation set (B, M, E, S) into two categories according to whether the label is a right-side boundary, we propose Eq. (10) to calculate the Information Entropy (IE) of these two categories using their edge probability.

$$H_{category}(c) = -\sum_{i=N,Y} (P_i(c) + \gamma)\log(P_i(c) + \gamma) \tag{10}$$

where c is the current observed character; $P_x(c)$ denotes the post-probability of character c being annotated as category (or label) x; $P_N(c) = P_B(c) + P_M(c)$; $P_Y(c) = P_E(c) + P_S(c)$; $\gamma(\gamma = 0.0001)$ is for the smoothness problem. The greater the value of $H_{category}(c)$ is, the higher the uncertainty confidence of c is.

Since the four types of labels are divided into two groups in the above method, it might miss some distribution information to a certain extent. Thus, we propose Eq. (11) to calculate the uncertainty confidence using four labels, which directly makes use of the IE of the four labels' edge probability. The value of the uncertainty confidence using Eq. (10) and Eq. (11) is from 0 to 1.

$$H_{label}(c) = -\sum_{i=B,E,M,S} (P_i(c) + \gamma)\log(P_i(c) + \gamma) \tag{11}$$

4.2.2 The Context Diversity

In the active learning procedure, the context diversity of the character is as important as the uncertainty confidence of the character. For microblogs, since the amount of unlabelled microblogs is too large and there are a lot of partially overlapped samples, the measurement of the context diversity is more difficult.

Aiming to efficiently select samples with high annotation values from raw microblogs, we propose λ-active learning approach, which introduces the parameter λ to control the number of repeatedly selected partially overlapped sentences. We present Eq. (12) to measure the diversity of character boundary, which takes advantage of the nearest one character in the context of the current observed character.

$$F(c) = -(\frac{d^t(c)}{\lambda})^3 \tag{12}$$

where c is the current observed character; $d^t(c)$ indicates the frequency of t, which is the context of c, $d^t(c)$ is initially set to 0 and updates as $d^t_{i+1}(c) = d^t_i(c) + 1$ when t presents as the context of c for the $(i+1)^{th}$ time. This method controls the number of repeatedly selected samples by parameter λ. When $d^t(c) < \lambda$, the diversity value of c is from -1 to 0, which has very little effect on the annotation value of c. With the increase of $d^t(c)$, the decrease of $F(c)$ will speed up, leading to the increasing influence on the annotation value of c. When λ is set to different values, the curves of the diversity are shown in Fig. 2.

4.2.3 The Evaluation of Overall Annotation Values

After obtaining the uncertainty confidence and the context diversity of characters, we propose Eq. (13) to calculate the annotation value for a character.

$$\varphi_c(c) = \alpha H(c) + \beta F(c) \tag{13}$$

where $H(c)$ denotes the uncertainty confidence of c which

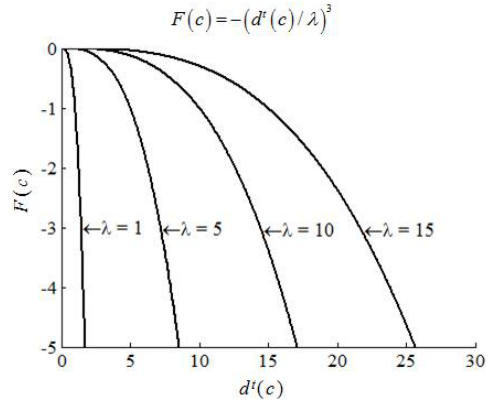

Fig. 2 The impact of parameter λ on the context diversity

can be calculated in two ways ($H_{category}(c)$ and $H_{label}(c)$); $F(c)$ denotes the context diversity of c; α and β are the weight of $H(c)$ and $F(c)$, respectively.

For CWS tasks, the annotation value of the whole sentence is more considerable than the annotation value of a single character. To select samples with high annotation values, three strategies are introduced to compute the overall annotation value of sample S. The results of these strategies are compared in the section of experiment analysis.

Avg-based Strategy: to adopt the average of the annotation values of all characters in the sample S as the annotation value of S.

Max-based Strategy: to adopt the maximum annotation value among all characters' annotation values in the sample S as the annotation value of S.

AvgMax-based Strategy: to combine both the average and maximum annotation values of all characters in the sample S as the annotation value of S.

4.3 λ-Active Learning Algorithm

During the corpus expansion process, the original training corpus is used to train the initial segmenter, which is applied to label the unlabeled microblogs. Then the uncertainty confidence and the context diversity of the character are obtained, and the annotation values of the characters and samples are subsequently evaluated by using the methods we proposed. The corpus are iteratively extended according to Algorithm 2.

Algorithm2 λ-active learning based corpus expansion algorithm

Input: the original training corpus $Train_0$, unlabeled samples $Unlabel_0$, the initial segmenter (CRFs), stopping condition D, iterator flag i = 0.
Output: the extended training corpus.
while True **do**
  Using $Train_i$ to train the CRFs model, and get $Model_i$.
  if the number of samples in $Train_i$ reaches D **do**
    break
  end if
  Use $Model_i$ to label the samples in $Unlabel_i$.
  for sample S in $Unlabel_i$ **do**
    Calculate $\varphi(S)$ of sample S.
  end for
  Sort samples in $Unlabel_i$ according to $\varphi(S)$.
  Select top M samples and modify their tags artificially.
  Put the selected samples into $Pool_i$.
  Delete samples in $Pool_i$ from $Unlabel_i$ to get $Unlabel_{i+1}$.
  Add samples in $Pool_i$ into $Train_i$ to get $Train_{i+1}$.
end while

5. Experiments and Results Analysis

5.1 Datasets

The training and test corpora are released by NLPCC 2015 for the shared task of microblog-oriented CWS [8], as shown in Table 2. In addition, we collect 300,000 unlabeled tweets (including 20 billion words) as the background corpus to extract features for the semi-supervised initial segmenter.

Table 2 Statistical information of datasets

Dataset	Sentences	Words	Characters
Training	10,000	215,027	347,984
Test	5,000	106,327	171,652
Total	15,000	322,410	520,555

Table 3 Best results of different sample selecting strategies

systems	P	R	F1
Baseline	93.46	92.99	93.22
Baseline_{PMI+AP}	94.02	93.71	93.87
Our_{Avg}	94.97	94.25	94.61
Our_{Max}	95.06	94.36	**94.71**
Our_{AvgMax}	95.05	94.33	94.69
WBA	95.01	94.34	94.67
Random	94.74	94.03	94.39
CRF++	93.3	93.2	93.3
Qiu 2013 [28]	94.1	93.9	94.0

5.2 Evaluation Metric

The segmentation results are evaluated by precision (P), recall (R), and F1-value (F1), which are defined as follows:

P = the number of correct tokens in prediction set / the number of all tokens in prediction set;

Recall = the number of correct tokens in prediction set / the number of correct tokens in standard set;

F1 = (2PR)/(P+R).

5.3 Results of λ-Active Learning Approach

To verify the effectiveness of the λ-active learning approach, several groups of experiments are conducted to assess the initial segmenters, the various sample-selected strategies and the value of parameter λ. The experimental results are shown in Table 3. The number of selected samples is same (500 tweets) in the iteration for all expanding strategies.

Baseline: For training the initial segmenter, only the basic context characters of the current observed character are used as context features for CRFs models and the size of

context window is empirically set as 5;

Baseline$_{PMI+AP}$: Except the basic context features, both the Point-wise Mutual Information (PMI) and the Anti-word Probability (AP) are also utilized as the semi-supervised features for CRFs models;

Our$_{Avg}$: For selecting samples from raw microblogs, Avg-based strategy is used;

Our$_{Max}$: Max-based strategy is used to select samples;

Our$_{AvgMax}$: AvgMax-based strategy is used to select samples;

WBA: For selecting samples, the state-of-the-art active learning method, WBA [23], is used;

Random: Samples are selected randomly.

According to the results in Table 3, the initial segmenter based on the semi-supervised feature which combines PMI and AP achieves the best performance, thus, we choose Baseline$_{PMI+AP}$ as the initial segmenter in the active learning procedure. We also see that among all the strategies we proposed, the Max-based strategy achieves the best results, with the AvgMax-based strategy following, and then the Avg-based strategy. All of the three strategies are obviously better than the Random-based strategy, which proves the effectiveness of the active learning approach. Besides, the data in Table 3 also shows that the F1-value of the Max-based λ-active learning method is higher than that of the state-of-the-art active learning method, for example WBA, which demonstrates that the λ-active learning method we proposed is more efficient in selecting samples from microblogs containing a large amount of partially overlapped sentences. Furthermore, we can also see that the result of our method is better than the outstanding previous works [28], as well as the Baseline$_{PMI+AP}$ with a gain of 0.84%, indicating that the Max-based λ-active learning method can prominently improve the progress of the CWS task.

Since parameter λ plays an important role in selecting partially overlapped samples from microblogs by controlling the number of repeatedly selected samples, we conduct the experiments while λ is set to various values, which is shown in Fig. 3. It is obvious that when λ is set to 5, the F1-value reaches the highest, which means the number of the partially overlapped samples added into the training corpus should be no more than 5.

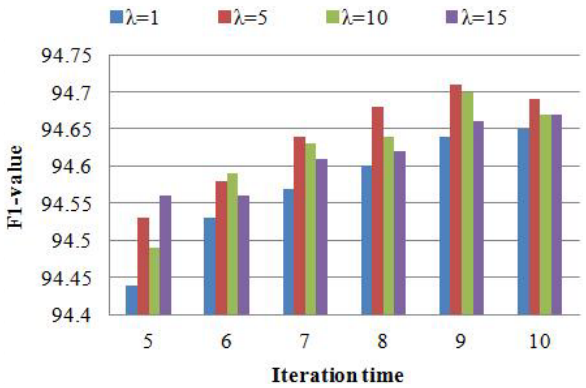

Fig. 3 The influence of parameter λ on CWS results

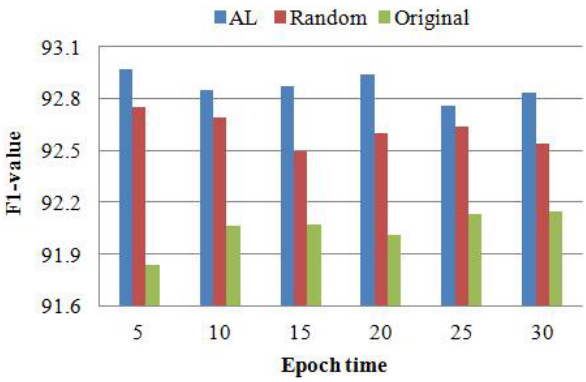

Fig. 4 Results of LSTM trained on different training corpus

5.4 Results of LSTM-Based CWS on Various Corpora

The extended training corpora are obtained by using our proposed Max-based λ-active learning methods combined with CRFs model as the initial segmenter. In order to assess the effectiveness of the extended training corpus on the LSTM neural networks, we conduct the following experiments. The experimental results are shown in Fig. 4.

Since the purpose of this paper is to research the effectiveness of the enlarged training corpus on the LSTM neural networks, the hyper-parameters of the LSTM neural networks were set as previous works as introduced in Sect. 3.

Original: The LSTM neural network is trained on the original training corpus with 10,000 tweets.

AL: The LSTM neural network is trained on the enlarged training corpus with 15,000 tweets as extended by our proposed λ-active learning method.

Random: For supervised machine learning methods, the scale of the training corpus usually has a strong impact on the performance of the models. Therefore, to be fair, we train the LSTM neural network on another enlarged training corpus which is extended randomly and contains 15,000 tweets.

In Fig. 4, the results show that the F1-value of AL is generally higher than that of Random, which means that the corpus expansion method we proposed can provide training corpus with more annotation values for the LSTM neural networks than randomly picked corpus. Furthermore, comparing to the Original, the improvement of AL is significant although the corpus we manually corrected is only 5,000, half the size of the original corpus, indicating that the LSTM neural networks trained on larger corpora with effective samples can achieve promising improvement on the performance of the microblog-oriented CWS task.

In our LSTM-based CWS architecture, we use Viterbi algorithm to choose the best path. The F1-values of our architecture on three types of corpora are shown as the group of "LSTM+VTB" in Fig. 5, which are obviously higher than that of the "LSTM" group which directly takes the outputs

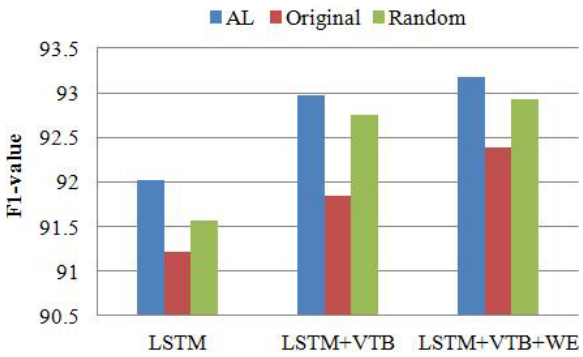

Fig. 5 Performance of different CWS architectures

of the LSTM neural network as the final results without using Viterbi algorithm. The results denote that the Viterbi algorithm plays a very important role in selecting labels according to the overall cost of the path.

In our CWS architecture, the character embeddings which are used to feed the LSTM neural networks are randomly generated during the training procedure. Since the scale of the training corpora can definitely affect the quality of character embeddings, we conduct one more group of experiments to avoid the impact of the corpora scale on training the character embeddings. We collect 300 million unlabeled tweets to pre-train the character embeddings using word2vec, and then utilize the pre-trained character embeddings to feed the LSTM neural networks. The F1-values are shown as the group of "LSTM+VTB+WE" in Fig. 5. As we can see, the improvement of λ-active-learning-based training corpus (AL) is still significant comparing with the original training corpus (Original) as well as the randomly selected corpus (Random). This is further evidence that the corpus extending method we proposed is effective for optimizing the performance of LSTM neural networks.

6. Conclusions

Aiming to improve the performance of the Neural CWS on microblog datasets, we treat CWS as a sequence labeling problem and propose a novel corpus expansion approach, based on the active learning methods with the CRFs model as the initial segmenter. Due to the partially overlapped sentence common in microblogs, parameter λ is introduced to the measurement of the context diversity and control the number of repeatedly selected samples in the active learning procedure. In this research, we first use Max-based strategy to measure the overall annotation values for a sample and achieve the best F1-value. The CRFs model is chosen as the initial segmenter to select samples for LSTM neural networks to avoid the parameter tuning and training. The experiment results show the training set enlarged by our method significantly improves the segmentation performance of LSTM neural networks on the microblog dataset.

In future research, we would like to further improve the segmentation performance of neural networks on microblog datasets by automatically generating pseudo-training corpus. We also want to combine the corpus expansion algorithm we proposed with the discriminator in the architecture of Generative Adversarial Networks (GANs).

Acknowledgments

This work has been supported in part by the National Natural Science Foundation of China under Grant 61672127. The first author Jing Zhang wishes to thank Professor Jianjun Ma, Yu Wang, Doctor Zhenchao Jiang, and Chen Liang for their useful suggestions, comments, and help during the design and editing of the manuscript.

References

[1] N. Xue and L. Shen, "Chinese word segmentation as LMR tagging," Proceedings of the second SIGHAN workshop on Chinese language processing - Volume 17, Sapporo, Japan, pp.176–179, 2003.
[2] D. Huang and D. Tong, "Context Information and Fragments Based Cross-Domain Word Segmentation," China Communications, vol.9, no.3, pp.49–57, 2012.
[3] X. Zheng, H. Chen, and T. Xu, "Deep Learning for Chinese Word Segmentation and POS Tagging," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), ACL, Seattle, Washington, USA, pp.647–657, 2013.
[4] W. Pei, T. Ge, and B. Chang, "Max-Margin Tensor Neural Network for Chinese Word Segmentation," Proceedings of the 52nd Annual Meeting of the ACL, Baltimore, Maryland, pp.293–303, 2014.
[5] J. Xu and X. Sun, "Dependency-based gated recursive neural network for Chinese word segmentation," Proceedings of the 54th Annual Meeting of the ACL, Berlin, Germany, pp.567–572, 2016.
[6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," Journal of Machine Learning Research, vol.12, pp.2493–2537, 2011.
[7] J. Zhang, D. Huang, X. Han, and W. Wang, "Rules-based Chinese Word Segmentation on MicroBlog for CIPS-SIGHAN on CLP2012," Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing, ACL, Tianjin, China, pp.74–78, 2012.
[8] X. Qiu, P. Qian, L. Yin, S. Wu, and X. Huang, "Overview of the NLPCC 2015 Shared Task: Chinese Word Segmentation and POS Tagging for Micro-blog Texts," Proceedings of the 4th CCF Conference on Natural Language Processing and Chinese Computing, Springer International Publishing, Nanchang, China, pp.541–549, 2015.
[9] X. Qiu, P. Qian, and Z. Shi, "Overview of the NLPCC-ICCPOL 2016 shared task: Chinese word segmentation for micro-blog texts," Proceedings of the Fifth Conference on Natural Language Processing and Chinese Computing & The Twenty Fourth International Conference on Computer Processing of Oriental Languages (NLPCC-ICCPOL 2016), Springer International Publishing, Kunming, China, vol.10102, pp.901–906, 2016.
[10] J. Eisenstein, "What to do about bad language on the internet," Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, Georgia, USA, pp.359–369, 2013.
[11] T.H. Nguyen and K. Shirai, "Topic modeling based sentiment analysis on social media for stock market prediction," Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp.1354–1364, 2015.
[12] X. Liu, M. Zhou, R. Wei, Z. Fu, and X. Zhou, "Joint inference of named entity recognition and normalization for tweets," Proceedings

of the 50th Annual Meeting of the ACL, Jeju Island, Korea, pp.526–535, 2012.
[13] C. Li and Y. Liu, "Improving Named Entity Recognition in Tweets via Detecting Non-Standard Words," Proceedings of the 53rd Annual Meeting of the Association for Computational Linguis- tics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp.929–938, 2015.
[14] J.-L. Lu, M.P. Kato, T. Yamamoto, and K. Tanaka, "Entity Identification on Microblogs by CRF Model with Adaptive Dependency," IEICE Transactions on Information & Systems, vol.E99-D, no.9, pp.2295–2305, 2016.
[15] A.N. Chy, M.Z. Ullah, and M. Aono, "Microblog Retrieval Using Ensemble of Feature Sets through Supervised Feature Selection," IEICE Transactions on Information & Systems, vol.E100-D, no.4, pp.793–806, 2017.
[16] M. Tang, X. Luo, and S. Roukos, "Active learning for statistical natural language parsing," Proceedings of the 40th Annual Meeting of ACL, philadelphia, Pennsylvania, USA, pp.120–127, 2002.
[17] S. Li, Y. Xue, Z. Wang, and G. Zhou, "Active Learning for Cross-domain Sentiment Classification," Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, pp.2127–2133, 2013.
[18] Y. Chen, T.A. Lasko, Q. Mei, J.C. Denny, and H. Xu, "A study of active learning methods for named entity recognition in clinical text," Journal of Biomedical Informatics, vol.58, pp.11–18, 2015.
[19] X. Chen, X. Qiu, C. Zhu, and X. Huang, "Gated recursive neural network for Chinese word segmentation," Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, pp.1744–1753, 2015.
[20] X. Chen, X. Qiu, C. Zhu, P. Liu, and X. Huang, "Long short-term memory neural networks for Chinese word segmentation," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Lisbon, Portugal, pp.1197–1206, 2015.
[21] Q. Zhou, L. Ma, Z. Zheng, Y. Wang, and X. Wang, "Recurrent neural word segmentation with tag inference," Proceedings of the China National Conference on Chinese Computational Linguistics, Springer International Publishing, Yantai, China, vol.10102, pp.734–743, 2016.
[22] D. Cai and H. Zhao, "Neural Word Segmentation Learning for Chinese," Proceedings of the 54th Annual Meeting of the ACL, Berlin, Germany, pp.409–420, 2016.
[23] S. Li, G. Zhou, and C.R. Huang, "Active learning for Chinese word segmentation," Proceedings of the 24th International Conference on Computational Linguistics, Mumbai, India, pp.683–692, 2012.
[24] X. Liang and L. Gu, "Active Learning in Chinese Word Segmentation Based on Nearest Neighbor," Computer Science, vol.42, no.6, pp.228–232, 2015.
[25] C. Feng, Z. Chen, H. Huang, and Z. Guan, "Active Learning in Chinese Word Segmentation Based on Multi-gram Language Model," Journal of Chinese Information Processing, vol.20, no.1, pp.52–60, 2006.
[26] Y. Wang, J'I. Kazama, Y. Tsuruoka, W. Chen, Y. Zhang, and K. Torisawa, "Improving Chinese Word Segmentation and POS Tagging with Semi-supervised Methods Using Large Auto-Analyzed Data," Proceedings of the Fifth International Joint Conference on Natural Language Processing, Chiang Mai, Thailand, pp.309–317, 2011.
[27] W. Sun and J. Xu, "Enhancing Chinese word segmentation using unlabeled data," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), ACL, Edinburgh, Scotland, UK, pp.970–979, 2011.
[28] X. Qiu, Q. Zhang, and X. Huang, "FudanNLP: A Toolkit for Chinese Natural Language Processing," Proceedings of the 51st Annual Meeting of the ACL, Sofia, Bulgaria, pp.49–54, 2013.

Jing Zhang received the B.S. degree in network engineering from Dalian Maritime University, Dalian, in 2011. She was jointly educated at center of computation learning systems in Columbia University, New York, USA, from 2014 to 2015. She is currently pursuing the Ph.D. degree in computer application technology at Dalian University of Technology. Her research interests include new word extraction from microblog-oriented data, Chinese word segmentation, social event extraction, question answering, and sentiment analysis in the field of natural language processing.

Degen Huang received the B.S. degree in computer science from Fuzhou University, China, in 1986, and the M.S. degree in computer software from the Dalian University of Technology, China, in 1988. He is currently a Professor with the School of Computer Science, Dalian University of Technology. His research interests include natural language processing, machine learning, and machine translation. He is now a senior member of CCF, CIPS, ACM, CAAI and an associate editor of Int. J. Advanced Intelligence.

Kaiyu Huang received the B.S. degree in computer science and the B.A. degree in Japanese from Dalian University of Technology, Dalian, in 2016. He is currently pursuing the master degree in computer application technology at Dalian University of Technology. His research interests include Chinese word segmentation and the sentiment classification in natural language processing.

Zhuang Liu received the B.S. and M.S. degree in Computer Science from Liaoning University in 2004 and 2007, respectively. He is currently pursuing the Ph.D. degree in computer application technology at Dalian University of Technology. His research interests include deep learning and natural language processing.

Fuji Ren was born in 1959. Ph.D. He is currently a Professor with Department of Information Science & Intelligent Systems, Tokushima University, 2-1 Minamijosanjima, Tokushima, 770-8506 Japan. His current research interests include natural language processing, machine translation, artificial intelligence, language understanding and communication, multi-lingual multi-function multi-media intelligent systems, super-function methodology.