New Word Detection and Tagging on Chinese Twitter Stream

Yuzhi Liang*, Pengcheng Yin*, and S.M. Yiu

Department of Computer Science
The University of Hong Kong
{yzliang,pcyin,smyiu}@cs.hku.hk

Abstract. Twitter becomes one of the critical channels for disseminating up-to-date information. The volume of tweets can be huge. It is desirable to have an automatic system to analyze tweets. The obstacle is that Twitter users usually invent new words using non-standard rules that appear in a burst within a short period of time. Existing new word detection methods are not able to identify them effectively. Even if the new words can be identified, it is difficult to understand their meanings. In this paper, we focus on Chinese Twitter. There are no natural word delimiters in a sentence, which makes the problem more difficult. To solve the problem, we derive an unsupervised new word detection framework without relying on training data. Then, we introduce automatic tagging to new word annotation which tag the new words using known words according to our proposed tagging algorithm.

Keywords: Chinese tweets · new word detection · annotation · tagging

1 Introduction

New social media such as Facebook or Twitter becomes one of the important channels for dissemination of information. Sometimes they can even provide more up-to-date and inclusive information than that of news articles. In China, Sina Microblog, also known as Chinese Twitter, dominates this field with more than 500 million registered users and 100 million tweets posted per day. An interesting phenomenon is that the vocabularies of Chinese tweets thesaurus have already exceeded traditional dictionary and is growing rapidly. From our observation, most of the new words are highly related to hot topics or social events. For example, the new word "*Yu'e Bao*" detected from our experimental dataset is an investment product offered through the Chinese e-commerce giant Alibaba. Its high interest rate attracted hot discussion soon after it first appeared, and without any concrete marketing strategy, *Yu'e Bao* has been adopted by 2.5 million users who have collectively deposited RMB 6.601 billion ($1.07 billion) within only half a month.

Obviously, these "*Tweet-born*" new words in the Chinese setting are worthy of our attention. However, finding new words from Chinese tweet manually is

* These two authors contributed equally to this work.

unrealistic due to the huge amount of tweets posted every day. It is desirable to have an automatic system to analyze tweets. The obstacle is that Twitter users usually invent new words using non-standard rules that appear in a burst within a short period of time. Existing new word detection methods are not able to identify them effectively. Even if the new words can be identified, it is difficult to understand their meanings.

In this paper, we focus on Chinese Twitter. There are no natural word delimiters in a sentence, which makes the problem more difficult. To solve the problem, we introduce a Chinese new word detection framework for tweets. This framework uses an unsupervised statistical approach without relying on hand-tagged training data for which the availability is very limited. Then, about new word interpretation, we proposed a novel method which introducing automatic tagging to new word annotation. The tagging results represent a big step towards automatic interpretation of these new words. Such kind of tagging is not only useful in new word's interpretation, but can also help other NLP tasks such as improving machine translation performance. Our proposed approach differs from existing solutions in the following ways.

1.1 An Unsupervised Statistical Method for Detecting Out-of-Vocabulary (OOV) Words in Chinese Tweets

Unlike English and other western languages, many Asian languages such as Chinese and Japanese do not delimit words by spaces. An important step to identify new words in Chinese is to segment a sentence into potential word candidates. Existing approaches to Chinese new word detection fall roughly into two categories: supervised (or semi-supervised) method and unsupervised method.

Supervised methods transform new word detection to a tagging problem and train the classifier based on tagged training sets. Both of the two most widely used Chinese word segmentation/new word detection tool Stanford Chinese-word-segmenter (based on Conditional Random Field CRF [16]) and ICTCLAS (based on Hierarchical Hidden Markov model HHMM[12]) are using supervised method. The problem is, precision of supervised method often relies on the quality of tagged training set. Unfortunately,there is no high quality tagged dataset specifically designed for Chinese tweets so far. Meanwhile, traditional training sets cannot capture all the features of microblog crops because microblog tweets are short, informal and have multivariate lexicons. Existing solutions [10][11] for identifying new words specially designed for the Chinese Microblog word segment are still supervised machine learning methods. Thus, both suffer from the shortage of good training datasets. Unsupervised method performs Chinese word segmentation by deriving a set of context rule or calculating some statistical information from the target data.

From our study, we notice that contextual rule-based approach is not suitable for the task of detecting new words from Chinese tweets because new words emerged from Sina Microblog are rather informal and may not follow these rules while statistical method is a good solution for this problem since it can be purely data driven. Our target is to define an efficient model to detect OOV

words from Chinese Twitter stream while avoiding using tagged datasets. We proposed a new word detection framework by computing the word probability of a given character sequence. This approach combines ideas from [3] [8] [9]. Detailed solution will be given in the later sections.

1.2 A Novel Method to Annotate New Word in Tweets by Tagging

Existing approaches for the new entity (phrase or words) interpretation include name entity recognition (NER) [5] and using the online encyclopedia as a knowledge base [2]. NER seeks to locate and classifies name entity into names of persons, organizations, locations, expressions of time, quantities, etc. [1]. However, this kind of classification cannot indicate the meaning of the new entity in detail. Another popular approach is interpreting entities by linking them to Wikipedia. This is not applicable for annotating new emerging words because most of new words will not have a corresponding/related entry in any online encyclopedias within a short period of time right after the new word's appearance.

To solve this problem, we propose a novel method which is annotating a new word by tagging it with known words. This is the first time word tagging is introduced to word annotation. Tagging is being extensively used in images (photos on facebook)[6] or articles annotation[7]. For new word tagging, the objective is to shed light on its meaning and facilitate users' better understanding. Our idea is to find out Top-K words that are most relevant to a given new word. The core issue of the problem is to derive a similarity measure between new words and their relevant words. Intuitively, words that co-occur with the new word with high frequency are more relevant with the new word. However, from our study, we found this naive definition might not be true for words in Chinese tweet. For instance, "*Mars brother*" is a nickname of "*Hua Chenyua*"(a singer) to indicate his abnormal behavior. These two terms are related but do not co-occur frequently in tweets because they can be a replacement of each other. Thus, we further quantify the similarity of two words by modeling the similarity of their corresponding contexts. The context of a word w is the surrounding text of w, roughly speaking, two words that share similar contexts are highly relevant. In this paper, we derived Context Cosine Similarity (CCS) which based on cosine similarity for the similarity measurement. The results show CCS can evaluate similarity between two words with high efficiency.

In our experiment, the approaches are evaluated with real microblog data for 7 consecutive days (2013-07-31 to 2013-08-06), which contains 3 million tweets in total. We compare our OOV detection approach with that of the most popular Chinese word segmentation tools ICTCLAS and Stanford Chinese-word-segmenter. The results show our method is competitive in OOV detection regarding to precision and recall rate. In new word tagging, we measure the accuracy of our tagging method by checking the existence of the generated tag words in corresponding Baidu Entry (Baidu Entry is an online encyclopedia like Wikipedia). The average precision is as high as 79%.

2 New Word Detection

2.1 Definition of New Word

In order to get the new word set $S_{word}(t-t_0)$ which contains new words appears at time t but not exists at time $t_0(t_0 < t)$, we need to get the word set at time t_0 ($S_{word}(t_0)$) and the word set at time t ($S_{word}(t)$) from unsegmented tweets at time t ($S_{tweet}(t)$). For any word w extracted from $S_{tweet}(t)$, if $w \in S_{word}(t)$ and $w \notin S_{word}(t_0)$, w is regarded as a new word, otherwise w is regarded as a known word.

2.2 Word Extraction

For a set of unsegmented tweets $S_{tweet} = \{T_1, T_2, ..., T_N\}$, the first step is to extract word segments in S_{tweet}. We have discussed in the introduction that the state-of-art supervised method is not suitable for our application due to the lack of training corpus. Instead of relying on training dataset, we take a statistical approach. It is worthy to notice that Symmetrical Conditional Probability (SCP) [8] is to measure the cohesiveness of a given character sequence while Branch Entropy (BE) [9] is to measure its extent of variance. These two statistical approaches measure the possibility of s being a valid word from two perspectives such that they can complement each other in achieving accuracy. Also, we use a word statistical feature Overlap Variety [3] to further reduce the noise. For each $T \in S_{tweet}$, a probability score will be calculated for all the consecutive character sequences with length between two and four in T to measure how likely the character sequence s is a valid word based on above features.

Sequence Frequency Sequence Frequency is an important noise filtering criteria in Chinese word segmentation. It is base on the concept that if s is a valid word, it should appear repeatedly in S_{tweet}. In this study, we only consider words with $freq(\cdot)$ larger than certain threshold T_{freq}, which is set to 15[1].

Symmetrical Conditional Probability Symmetrical Conditional Probability (SCP) is defined to measure the cohesiveness of a given character sequence s by considering all the possible binary segmentations of s. Let n denotes the length of s, c_x denotes the x^{th} character in s, $P(\cdot)$ denotes the possibility of the given sequence appearing in the text, which is estimated by its frequency, the SCP score of s $SCP(s)$ is as 1:

$$SCP(s) = \frac{P(s)^2}{\frac{1}{n-1}\sum_{i=1}^{n-1} P(c_1,c_i)P(c_{i+1},c_n)} \quad (1)$$

[1] Here 15 is an experimental number, but this number can be evaluated by some statistical features such as mean and standardization of all the character sequences' frequency

Branching Entropy Symmetrical Conditional Probability (SCP) is defined to measure the cohesiveness of a given character sequence s by considering all the possible binary segmentations of s. Let n denotes the length of s, c_x denotes the x^{th} character in s, $P(\cdot)$ denotes the possibility of the given sequence appearing in the text, which is estimated by its frequency, the SCP score of s $SCP(s)$ is as 2:

$$H(x|s) = -\sum_{x \in X} P(x|s) \log P(x|s) \tag{2}$$

We denote this $H(x|s)$ as $H_L(s)$ such that $H_R(s)$ can be defined similarly by considering the character following s. The Branch Entropy of sequence s is as 3:

$$BE(s) = \min \{H_L(s), H_R(s)\} \tag{3}$$

Word Probability Score First of all, character sequences have extremely low BE score or SCP score will be abandoned and character sequences have extremely high BE score or SCP score can be selected as valid word directly. For the rest of words, we define an word probability score $Pr_{word}(s)$ to indicates how likely a given character sequence s is a valid word. $Pr_{word}(s)$ is calculated based on normalized BE and SCP of s as 4

$$Pr_{word}(s) = (1 + \mu)Nor(BE(s)) + Nor(SCP(s)) \tag{4}$$

We added μBE(s) in calculating the word probability because we found BE score is more important than SCP score when defining whether character sequence s is a valid word. We set μ to 0.2 in our experiment. $Nor(BE(s))$ is the normalized BE score of s which used max-min method to perform the normalization:

$$Nor(BE(s)) = \frac{BE(s) - Min_{BE(s)}}{Max_{BE(s)} - Min_{BE(s)}} \tag{5}$$

$Nor(SCP(s))$ is the normalized SCP score of s. Experimental result shows that SCP scores of the character sequences are very uneven, so shift z score mentioned in [12] which can provide an shift and scaling zscore to normalize the majority of SCP score into [0,1] is used to perform the SCP score normalization:

$$Nor(SCP(s)) = \frac{\frac{SCP(s)-\mu}{3\sigma} + 1}{2} \tag{6}$$

Character sequence with word probability score larger than a certain threshold will be considered as a valid word. From our observation, most character sequences with $Pr_{word}(\cdot)$ larger than 0.3 are valid words. This threshold also can be evaluate with some statistical features such as mean and standard derivation of Pr_{word} for all the character sequences with length between 2 to 4.

Noise Filtering Although we can get valid word candidate by setting a threshold on $Pr_{word}(\cdot)$, substring of a valid word exists as noise with relative high $Pr_{word}(s)$ from our observation. A dictionary will be used as knowledge base in noise filtering. The basic idea to filter this kind of noise is to consider word probability of the given character sequence and its overlapping strings [3]. For a candidate word w_0, assume its left overlapping string s_L is defined as $c_1c_2...c_kc_{k+1}...c_l, l = \{2, ..., |w_0|\}$ where $c_1...c_k$ is the k-character sequence proceeding of w_0 and $c_{k+1}...c_l$ is the first $l-k$ characters of s. Right overlapping string is defined similarly. For a noise word, it always has some overlapping strings s with $Pr_{word}(s) > Pr_{word}(w_0)$. For example, $Pr_{word}(\text{中国}) > Pr_{word}(\text{国媒})$ (because 中国, China, is a dictionary word and $Pr_{word}(\text{中国}) = 1$ while 国媒is a wrong segment). But for a valid word, mostly $Pr_{word}(w_0)$ is larger than $Pr_{word}(s)$. For each selected candidate words w_0, let $S_{ov}(w_0)$ denotes the set of overlapping sting of w_0, w_0's overlapping score $OV(w_0)$ is then calculated as follows:

$$OV(w_0) = \frac{\sum_{s \in S_{ov}(w_0)} I(Pr_{word}(w_0) > Pr_{word}(s))}{|S_{ov}(w_0)|} \tag{7}$$

$I(\cdot)$ is the indicator function. Candidate words with $OV(\cdot)$ larger than certain threshold are rejected.

Pseudo code of new word detection process is stated in **Algorithm 1**.

Algorithm 1 New Word Detection

```
Let T_freq, T_pr and T_ov denotes the thresholds of frequency, word probability score and overlapping score respectively
for all character sequence s, 2 ≤ |s| ≤ 4, s is substring of T, T ∈ S_tweet(t) do
    Count occurrences of s freq(s) in S_tweet(t)
    if freq(s) ≥ T_freq then
        Compute SCP(s) and BE(s) using formula 1 and 3
        Compute Pr_word(s) based on BE(s) and SCP(s) using formula 4
        if Pr_word > T_Pr then
            Add s to word candidate set Cand(s)
        end if
    end if
end for
for all s ∈ Cand(s) do
    Compute OV(s)
    if OV(s) < T_ov then
        if s ∈ S_tweet(t_0) then
            Add s to S_word(t_0) (S_word(t_0) is the known word set)
        else
            Add s to S_word(t − t_0) (S_word(t − t_0) is the new word set)
        end if
    end if
end for
```

Here we first select all the frequent character sequences, then calculate word probability of these character sequences bases on their SCP and BE score. Finally, we filter noise by applying 7 to get the valid words.

3 New Word Tagging

Regarding to new word interpretation, our method introducing tagging by tagging new word $w_{new}(w_{new} \in S_{word}(t - t_0))$ with known word $w_{known}(w_{known} \in S_{word}(t_0))$. Words in the following two categories are potential tag words:

- Words that are highly relevant to w_{new}, i.e. its attributes, category and related Named Entities.
- Words that are semantically similar to w_{new}, i.e. synonyms.

The First category of words is important for tagging new words related to certain social events. It may include the event's related people, institutions, and microblog user's comments. Those words co-occur with w_{new} frequently. For the second category, w_{new}'s synonyms may not co-occur with w_{new} as mentioned in the previous example "*Mars brother*" and "*Hua Chenyu*". It is obvious that approaches such as picking words that co-occur with w_{new} as tagging words is rather naive. We seek to develop a similarity measure between two words that not only incorporate word co-occurrence, but can also utilize other features to deal with the above case. We found that for w_{new} and its potential tagging word w_{known}, no matter which category w_{known} belongs to, it shares similar context with w_{new}. A word w_{known}'s context is its surrounding text which may shed light on its meaning. We could simply model w_{known}'s context as the set of words co-occurring with w_{new} in $S_{tweet}(t)$.

3.1 Context Cosine Similarity

Given a new word w_{new}, the basic idea of tagging w_{new} is to find its most similar known words w_{known}. The amount of tweets is huge even just about a single topic, and the contents of tweets often cross domains. According to these characteristics, we decide to use cosine similarity to perform the similarity measurement for its efficient and domain independent. The context cosine similarity between a new word w_{new} and a known word w_{known} is computed as follow:

1. Let $D(w_1, w_2)$ denotes the pseudo document made by concatenation of all tweets containing w_1 while w_1, w_2 are excluded from the document. Compute $D(w_{new}, w_{known})$ and $D(w_{known}, w_{new})$.
2. Compute V_{new} and V_{known} where $V = \{V_1, V_2, ..., V_n\}$ is the term vector of a Document D. V_i in V is the TF-IDF weight [2] of w_i, $w_i \in S_{word}(t_0)$ and $i = \{1, 2, ..., n\}$ where n is the size of $S_{word}(t_0)$

[2] TF-IDF is a numerical statistic used to indicate the importance of the given word in a corpus. The score is TF $\times$ IDF, where TF is term frequency which is a normalized term count, IDF is Inverse Document Frequency which indicates the proportion of documents in the corpus containing w_i.

3. Context cosine similarity between w_{new} and w_{known} is defined as

$$\begin{aligned} Sim_{rawccs}(w_{new}, w_{known}) &= \frac{V_{new} \cdot V_{known}}{|V_{new}||V_{known}|} \\ &= \frac{\sum_{i=0}^{n} V_{newi} \cdot V_{knowni}}{\sqrt{\sum_{i=1}^{n} V_{newi}^2} \times \sqrt{\sum_{i=1}^{n} V_{knowni}^2}} \end{aligned} \tag{8}$$

4. We get Sim_{ccs} by normalizing Sim_{rawccs} to [0,1] using max-min normalization[3] on the top 20 tag words of the new word.

Worth noting that in Step 1, we excluded w_{new} and w_{known} from $D(w_{new}, w_{known})$ and $D(w_{known}, w_{new})$ is because we assume if two words are semantically similar, their context should be similar even they co-occur with low frequency.

3.2 Choose Tag Word

When tagging a new word w_{new}, we will compute Context Cosine Similarity between w_{new} and all the known words in $S_{word}(t_0)$. For a known word w_{known}, if $Sim_{ccs}(w_{new}, w_{known})$ is larger than a threshold k, w_{known} will be selected as a tag word of w_{new}. The value of $Sim_{ccs}(w_{new}, w_{known})$ is in the range [0,1]. According to 4, we set the threshold k to 0.5 as a balance point of the number of selected tag words and tag word precision.

4 Experiment

4.1 Dataset Setting

In this experiment, we aim at detecting newly emerged words on a daily basis. Regarding to the definition of new words, for the target day t, $S_{tweet}(t)$ is the set of tweets published on that day. Tweets published in seven consecutive days, from July 31st, 2013 to Aug 6th, 2013 are used as our input. Meanwhile, we use the tweets of May 2013 as the known word set $S_{tweet}(t_0), t_0 < t$ which serves as knowledge base.

We perform cleaning on dataset used as $S_{tweet}(t)$, where hash tags, spam tweets, tweets only contains non-Chinese characters are rejected. **Table 1** shows the details of our dataset. And we store any character sequence with length between two and four in $S_{tweet}(t_0)$ to serve as the known word set $S_{word}(t_0)$ to ensure new words detected from $S_{tweet}(t)$ has never appeared in $S_{tweet}(t_0)$.

4.2 New Word Detection Result

In the new word detection experiment, we use ICTCLAS and Stanford Chinese-word-segmenter [16] to serve as our baselines. The training data used by ICTCLAS is Peking University dataset which contains 149,922 words while training

[3] $Sim_{ccs} = \frac{Sim_{rawccs} - Min_{rawccs}}{Max_{rawccs} - Min_{rawccs}}$

Table 1: List of dataset

Dataset	# of tweets	After cleaning
July 31	715,680	443,734
Aug 1	824,282	515,837
Aug 2	829,224	516,152
Aug 3	793,324	397,291
Aug 4	800,816	392,945
Aug 5	688,692	321,341
Aug 6	785,236	399,699
May	20,700,001	-

data used for CRF training is Penn Chinese Treebank which contains 423,000 words. All the words appearing in the training set will not be selected as new word. And our aim is to detect new words of certain importance as well as their relevant words, it is reasonable to focus on words with relatively high frequency. In this experiment, words appearing less than 15 times will be ignored. In addition, non-Chinese character, emotion icon, punctuation, date, word containing stop words and some common words are excluded because they are not our target.

Generally speaking, Chinese new words can be divided into several categories [15] (excluding new words with non-Chinese characters): name entity, dialect, glossary, novelty, abbreviation and transliterated words. The detected new words are classified according to these categories in our experiment. The precision of the detection result is defined as 9

$$Precision = \frac{\text{\# of valid new words}}{\text{\# of total new words dtected}} \tag{9}$$

The results are listed in **Table 2**.

Table 2: New word detection result

Category	Our method	ICTCLAS	Chinese-word-segmenter
Name Entity	50	36	60
Glossary	2	1	6
Novelty	19	2	36
Abbreviation of hot topic	22	1	8
Transliterated words	0	0	5
Noise	4	5	139
Valid new words	93	40	**115**
Precision	**95.9%**	88.9%	45.2%

The above results show that our method has the highest precision in detecting new words in Chinese Twitter among the three methods. Stanford Chinese-

word-segmenter wins in recall. However, a large number of noise is also included in Stanford Chinese-word-segmenter's result which lowers the precision tremendously. The reason is that it uses a supervised machine learning method, for which the shortage of appropriate tagged training dataset for Chinese tweet is a fatal problem. ICTALS has an acceptable precision, but it often over segment the words which makes it fails to detect some compound words such as "*Yu'E Bao*" and "*Micro-channel Voca*".

4.3 New Word Tagging Result

As stated above, words have context cosine similarity with a new word larger than a certain threshold are selected as the new word's tags. Words such as 加油(work hard) and 执行(operate) are excluded in tag words manually since they are either only popular in Sina Microblog or do not have much meaning on its own. Some tagging result examples are listed as below:

- **Liu Yexing** (Name Entity. A guy from Tsinghua University become famous by attending reality show "Who's still standing" and burst their question bank)
  Tags: Zhang Xuejian(Liu Yexing's adversary), Who's still standing, Tsinghua University, Peking University, question bank, answer questions
- **Ergosterol**(Glossary. Ergosterol is a sterol found in cell membranes of fungi and protozoa, serving many of the same functions that cholesterol serves in animal cells)
  Tags: protein, amount, growth, immunity, vegetables, growing development
- **Yu'E Bao**(Novelty. Yu'E Bao is a money-market fund promoted by Alipay)
  Tags: currency, Internet, finance, fund, money management, supply-chain
- **Burst Bar event**(Abbreviation of hot topic. Pan Mengyin, a fan of Korean star G-Dragon, spread some inappropriate remarks about football stars which makes fans of the football stars get angry and attacked G-Dragon's Baidu Bar.)
  Tags: Pan Mengyin, G-Dragon , Korean star, stupid

Here we set Context Cosine Similarity threshold to 0.5 such that we can get enough tag words while achieving a relatively high precision. **Table 3** shows the average number of tags and tagging precision using different similarity threshold. Among the 93 detected new words, some of them are recorded in Baidu Entry now. We randomly picked 20 recorded words from different categories to evaluate our tagging result. The precision of tagging result about a new word (w_{new}) is defined as:

$$Precision_{tag}(w_{new}) = \frac{\#\text{ of tag words hits in } w_{new}\text{'s Baidu Entry}}{\#\text{ of } w_{new}\text{'s tag words}} \tag{10}$$

From **Table 3** we can see that the number of selected tag words decreases while the tagging precision increase when Context Cosine Similarity threshold arise. That means tag word have higher context cosine similarity with the new word is more likely be the right tag of the new word.

Table 3: Word tagging result 1

Threshold	Average # of tags	Average precision
0	19.6	0.56
0.25	9.1	0.71
0.5	5.5	0.79
0.75	3	0.825

We also compared the number of tags and tagging precision of different word categories when the CCS threshold is 0.5(See **Table4**).

Table 4: Word tagging result 2

Category	# of new words	Average # of tags	Average precision
Name entity	9	6.11	0.80
Glossary	1	6.00	0.00
Novelty	4	3.00	0.96
Abbreviation of hot topic	6	6.17	0.79

An interesting phenomena is that comparing to name entity and abbreviation of hot topic, novelty have fewer number of tag words while achieves very high precision. And we failed to tag the glossary *ergosterol* precisely because a lot of tweets talking *ergosterol* are a kind of advertisement.

5 Conclusion and future work

In this paper, we consider the problem of detecting and interpreting new words in Chinese Twitter. We proposed an unsupervised new word detection framework which take several statistical features to derive a word probability score that can measure word-forming likelihood of a character sequence. Since this framework is a statistical approach, it could be easily applied to other languages that have similar characteristics as Chinese characters (e.g. No natural word delimiters).

Then, we used automatic tagging in new word interpretation. We derive a similarity measure between new word and its candidate tag word based on similarity of their corresponding contexts. Experiments on real datasets show the effectiveness of our approach. However,in this work, some thresholds, such as $freq(\cdot)$ and $Pr_{word}(s)$, are set by experiments and observation. In real practise, we can have a more systematic and statistical way to set some appropriate thresholds. For example, for the frequency, we can compute the mean and the standard deviation of the identified words, then set a threshold based on the mean and the standard deviation. In the future, we will try to explore an automatic way to define the parameters used in this framework and apply the language model in our research to get more accurate results.

References

1. Ritter, Alan, Sam Clark, and Oren Etzioni. "Named entity recognition in tweets: an experimental study." Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011.
2. Gattani, Abhishek, et al. "Entity extraction, linking, classification, and tagging for social media: a wikipedia-based approach." Proceedings of the VLDB Endowment 6.11 (2013): 1126-1137
3. Yunming Ye, Qingyao Wu, Yan Li, K. P. Chow, Lucas Chi Kwong Hui, and Siu-Ming Yiu. Unknown chinese word extraction based on variety of overlapping strings. Inf. Process. Manage., 49(2):497-512, 2013
4. Hai Zhao and Chunyu Kit. Exploiting unlabeled text with different unsupervised segmentation criteria for chinese word segmentation. Research in Computing Science, 33:93-104, 2008.
5. Nadeau, David, and Satoshi Sekine. "A survey of named entity recognition and classification." Lingvisticae Investigationes 30.1 (2007): 3-26.
6. Zhou, Ning, et al. "A hybrid probabilistic model for unified collaborative and content-based image tagging." Pattern Analysis and Machine Intelligence, IEEE Transactions on 33.7 (2011): 1281-1294.
7. Kim, Heung-Nam, et al. "Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation." Electronic Commerce Research and Applications 9.1 (2010): 73-83.
8. Luo, Shengfen, and Maosong Sun. "Two-character Chinese word extraction based on hybrid of internal and contextual measures." Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17. Association for Computational Linguistics, 2003.
9. Jin, Zhihui, and Kumiko Tanaka-Ishii. "Unsupervised segmentation of Chinese text by use of branching entropy." Proceedings of the COLING/ACL on Main conference poster sessions. Association for Computational Linguistics, 2006.
10. Wang, Longyue, et al. "CRFs-based Chinese word segmentation for micro-blog with small-scale data." Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language. 2012.
11. Zhang, Kaixu, Maosong Sun, and Changle Zhou. "Word segmentation on Chinese mirco-blog data with a linear-time incremental model." Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing, Tianjin, China. 2012
12. Zhang, Hua-Ping, et al. "HHMM-based Chinese lexical analyzer ICTCLAS." Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17. Association for Computational Linguistics, 2003 Aksoy, Selim, and Robert M. Haralick. "Feature normalization and likelihood-based similarity measures for image retrieval." Pattern Recognition Letters 22.5 (2001): 563-582.
13. Dumais, Susan T. "Latent semantic analysis." Annual review of information science and technology 38.1 (2004): 188-230.
14. Kityz, Chunyu, and Yorick Wilksz. "Unsupervised learning of word boundary with description length gain." Proceedings of the CoNLL99 ACL Workshop. Bergen, Norway: Association for Computational Linguis-tics. 1999.
15. Zou Gang , et al " Chinese New Words Detection in Internet" Chinese Information Technology 18.6 (2004): 1-9.
16. Tseng, Huihsin, et al. "A conditional random field word segmenter for sighan bakeoff 2005." Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing. Vol. 171. Jeju Island, Korea, 2005.