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, Pokfulam, 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 first introduce a method of detecting new words in Chinese twitter using a statistical approach without relying on training data for which the availability is limited. Then, we derive two tagging algorithms based on two aspects, namely word distance and word vector angle, to tag these new words using known words, which would provide a basis for subsequent automatic interpretation. We show the effectiveness of our algorithms using real data in twitter and although we focus on Chinese, the approach could be applied to other Kanji based languages.

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, which makes them appear repeatedly in different social media and people's daily life. 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

A. Hameurlain et al. (Eds.): TLDKS XXXII, LNCS 10420, pp. 69–90, 2017.
DOI: 10.1007/978-3-662-55608-5_4

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 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, such as [1,16], 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. The contributions of our paper are listed as below:

- We introduce a Chinese new word detection[1] framework for tweets. This framework uses an unsupervised statistical approach without relying on hand-tagged training data for which the availability is very limited. The proposed framework is compared with ICTCLAS and Stanford Chinese-word-segmenter on new word detection over real microblog data (2013-07-31 to 2013-08-06). The result shows our method is competitive in new word detection regarding precision and recall rate. Although we focus on Chinese, the approach could be applied to other Kanji based languages like Japanese or Korean.
- We propose a novel method to annotate Chinese new words in microblog by automatic tagging. Context Distance and Context Cosine Similarity are derived for the similarity measurement. To the best of our knowledge, it is the first time automatic tagging is used in word interpretation. The new word tagging result accuracy is measured by checking the existence of the generated tag words in corresponding Baidu Entry (Baidu Entry is an online encyclopedia like Wikipedia. Some new words are recorded in Baidu Entry serval months after its first appearance). The average precision of tagging by Context Distance and that of Context Similarity are 52% and 79% respectively.

2 Related Works

2.1 New Word Detection 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.

The core idea of supervised method is transferring the segmentation task to a tagging problem, each character is represented as a one-hot vector or an n-gram vector then by using a pre-trained classifier, a tag of the character will be generated to indicates the position of the character in a word (i.e. 'B' for beginning, 'M'

[1] Also known as Out-of-Vocabulary (OOV) detection.

for middle, ‘E’ for end, ‘S’ for single character as a word). Supervised method is popular in Chinese word segmentation. For example, [22] using a shallow (2 layers) neural network as a classifier to tag the characters, [1] using a discriminative undirected probabilistic graphical model Conditional Random Field in [23] to perform the classification. Moreover, both of the two most widely used Chinese word segmentation/new word detection tool Stanford Chinese-word-segmenter (based on Conditional Random Field CRF [21]) and ICTCLAS (based on Hierarchical Hidden Markov model HHMM [16]) are using supervised method. The problem is, precision of supervised method often relies on the quality of tagged training set. Unfortunately, as far as we know, the largest public hand-tagged training dataset of Chinese Microblog only consists of around 50,000 sentences, which is not sufficient to capture all features of Chinese new words in tweets. On the other hand, differ from other documents, microblog tweets are short, informal and have multivariate lexicons (words can form new words in various ways) which makes training sets of traditional documents is not so suitable for microblog crops. The difference between semi-supervised method and supervised method is that semi-supervised method (e.g. [7,9]) tries to derive statistical information from the training datasets such as Description Length Gain (DLG) [19] in which the best segmentation of a sentence is computed to maximize information gain. Although this information can be used to identify new words, the computation of training dataset features is time-consuming and the accuracy still relies on the quality of training datasets. Existing solutions [14,15] for identifying new words specially designed for Chinese Microblog word segment are also supervised machine learning methods. Thus, both suffer from the shortage of good training datasets.

Unsupervised method perform 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.

2.2 New Word Annotation

Existing approaches for the new entity (phrase or words) interpretation include name entity recognition (NER) [8] and using the online encyclopedia as a knowledge base [4]. NER seeks to locate and classifies name entity into names of persons, organizations, locations, expressions of time, quantities, monetary values, and percentages, etc. [2,3,9]. However, the new words we detect in Sina tweets are not limited in name entity. Some of them are new adjectives such as “坚韧淡定” (clam and tough). Even though NER can classify the new entity into different categories, the meaning of the new entity is still missing. 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 comes out.

3 Overview

In this paper, we describe an unsupervised Chinese new word detection technic in Sect. 4. There is no training set required by this method such that it is a good match for new word detection in Chinese tweets whose high-quality training set is unavailable. Then for new word interpretation, we propose a novel framework in Sect. 5 to realize new word annotation by automatic tag the new words with known words. Some notations in this paper is listed in Table 1.

Table 1. Notations

Symbol	Description
s	Character sequence
n	The number of characters in s
c_i	The i^{th} character in s
w	Word
w_{new}	New word
w_{known}	Known word
t	Target time point
t_0	A time point before t
D	Tweet corpus
T	A tweet
Set_{tweet}	Unsegment tweets
k	Number of words in Set_{known}
Set_{new}	New word set
Set_{known}	Known word set
Set_{cand}	Candidate word set
$doc(w_1, w_2)$	Document made by tweets containing w_1 while w_1 and w_2 are excluded from the document
$doc(w)$	Document made by all the tweets containing w

4 New Word Detection

In new word detection, 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 several unsupervised Chinese word segmentation methods, i.e. [6,12,13], as follows. In statistical Chinese segmentation methods, a common basic assumption is that a Chinese word should appear as a stable sequence in the corpus. Symmetrical

Conditional Probability (SCP) [12] can be used to measure the cohesiveness of a given character sequence. On the other hand, Branching Entropy (BE) [13] measures the extent of variance based on the idea that if a character sequence s is a valid word, it should appear in different contexts. We note that these two statistical approaches measure the possibility of s being a valid word from two perspectives. SCP captures the cohesiveness of the characters in s, while BE considers the outer variability. They can complement each other in achieving accuracy. To further reduce the noise, we use a word statistical feature Overlap Variety in [6].

4.1 Definition of New Word

New word of time t is defined as words appears at time t but not exists at time $t_0 (t_0 < t)$. In our study, we are only interested in words whose frequencies are larger than a certain threshold for the following reasons: Firstly, low-frequency character sequences usually are meaningless character sequences. Secondly, even some of them are valid words, they are mainly people names just known by the posters or misspelled words which are not our target. Thirdly, we need to annotate the new words after they are detected, low-frequency character sequence have not enough relevant data for automatic tagging. In the process of new word detection in Chinese tweets, firstly, we need to get the word set at time t_0 and the word set at time t from the unsegmented tweets. Then for any word w extracted from the unsegmented tweets at t $Set_{tweet}(t)$, if w is not exist in $Set_{tweet}(t_0)$, w is regarded as a new word, otherwise w is a known word.

4.2 Word Extraction

The first step is to extract word segments from a set of unsegmented tweets. 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 data, we propose an unsupervised approach for Chinese new word detection from tweets. Different statistical based Chinese word segmentation approaches are jointly used in the proposed method. Symmetrical Conditional Probability (SCP) [12] is a method which evaluates the cohesiveness of a character sequence while Branching Entropy (BE) [13] measures the environment variance of a character sequence. These two approaches can complement each other in achieving accuracy. Moreover, Overlap Variety [6] is further used in reducing noise. For each character sequence in the set of unsegmented tweets with a length between two and four, a probability score will be calculated to indicate how likely the character sequence is a valid word. Technical detail will be introduced in the following parts.

Sequence Frequency. Sequence Frequency is an important noise filtering criteria in Chinese word segmentation. It is base on the assumption that if s is a valid word, it should appear repeatedly in Set_{tweet}. In our study, character sequences whose frequency lower than a threshold, $Thres_{freq}$, are filtered beforehand.

Symmetrical Conditional Probability. Symmetrical Conditional Probability (SCP) is a statistical criterion which measures the cohesiveness of a given character sequence s by considering all the possible binary segmentations of s. It based on the assumption if s is a valid word, the substrings of s will mainly appear along with s. For example, given sentence “氨基酸/是/构成/蛋白质/的/基本/单位” (Amino acids constitute the basic unit of protein). The character sequence “氨基酸” (Amino acids) is a valid word, its substrings (i.e. “ 氨基”,“酸”,“氨”,“基酸”) should mainly co-occur with “氨基酸”.

Formally, let n denotes the length of s, c_i denotes the i^{th} character in s, the possibility of the given sequence appearing in the text, which is estimated by its frequency, the SCP score of s is by Eq. 1.

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

The value of $SCP(s)$ is in range $(-\infty, 1]$. $SCP(s)$ is high when all the binary segmentations of s mainly appear along with s.

Branching Entropy. Branching Entropy (BE) measures the extent of the variance of the contexts in which s appears. It is based on the idea that if s is a valid word, it should appear in different contexts. Branching Entropy quantifies such context variance by considering the variance of the preceding and following character of s. Assume X is the set of the preceding characters of s, $P(s|x)$ is the probability that x is followed by s, the formula is defined by Eq. 2.

$$P(s|x) = \frac{P(x+s)}{P(x)} \tag{2}$$

Here $P(x+s)$ is the frequency of character sequence $(x+s)$ and $P(x)$ is the frequency of x.

The Left Branching Entropy of $P(x|s)$ is defined as Eq. 3:

$$BE_{left}(s) = -\sum_{x \in X} P(s|x) \log P(s|x) \tag{3}$$

The Right Branching Entropy, $BE_{right}(s)$, can be defined similarly by considering the characters following s. The overall Branching Entropy of sequence s is defined by Eq. 4:

$$BE(s) = \min\{BE_{left}(s), BE_{right}(s)\} \tag{4}$$

The value of $BE(s)$ is in range $[0, \infty)$. $BE(s) = 0$ if the size of the preceding character set of s or that of the succeeding character set of s equals to 1. In other words, the value of $BE(s)$ is 0 when the left boundary or the right boundary of s occurs in only one environment in the given corpus. For a specific s, the value of $BE(s)$ is high if both of the size of the preceding character set and the succeeding character set is large which means s often happens in different context such that s is likely to be a valid word. For instance, given the character sequence “门把手(doorknob)” which is a valid word, we might find it in different contexts such as “门把手坏了(The doorknob is broken)”, “要一个新的门把手(Need a new doorknob)”, “或者把这个门把手修好(Or repair this doorknob)”, “这个门把手很漂亮(This doorknob is pretty)”. In this case, there are three different preceding characters of “门把手 (sentence start, ‘的’, ‘个’)”, as well as four different succeeding characters (‘坏’, sentence end, 修’, ‘很). The Left Branching Entropy of “门把手” is $BE_{left}(r) = -(\frac{1}{4} \times log(\frac{1}{4}) + \frac{1}{4} \times log(\frac{1}{4}) + \frac{1}{2} \times log(\frac{1}{2})) = 1.5$, the Right Branching Entropy of “门把手(doorknob)” is $BE_{right}(r) = -(\frac{1}{4} \times log(\frac{1}{4}) \times 4) = 2$, the overall Branching Entropy takes the minimum of these two values, i.e.,1.5. On the other hand, for an invalid character sequence “门把” (a subsequence of the valid word “门把手”), the number of different preceding of “门把” is also three (sentence start, “的’, ’个”), but the number of succeeding of “门把” is just one (‘个’), so $BE_{left}(r) = 1.5$ while $BE_{right}(r) = -(1 \times log(1)) = 0$, and the overall BE value of “门把” is 0.

Word Probability Score. In the process of finding valid word from the set of character sequences, the character sequences have extremely low BE score or SCP score will be abandoned and the character sequences whose BE score or SCP are larger than the corresponding threshold can be selected as valid word directly beforehand. Then an word probability score $P_{word}(s)$ is defined for the reset of character sequences to indicates how likely a character sequence s is a valid word. The score is calculated based on normalized BE and SCP of s. s is probably a valid word if $P_{word}(s)$ is high. The formula of $P_{word}(s)$ is as Eq. 5.

$$P_{word}(s) = w_{BE} \times BE'(s) + w_{SCP} \times SCP'(s) \tag{5}$$

$BE'(s)$ is the normalized BE score of s which is defined by max-min normalization of the BE values Eq. 6

$$BE'(s) = \frac{BE(s) - min_{BE}}{max_{BE} - min_{BE}} \tag{6}$$

The value of $BE'(s)$ is in range [0,1].

$SCP'(s)$ is the normalized the SCP score of s. Experimental result shows that SCP scores of the character sequences are not normally distributed, the commonly used normalization method max-min normalization and z-score normalization is not efficient in this case. We use a shift z-score mentioned in [16] which can provide an shift and scaling z-score to normalize the majority of the SCP values into range [0, 1]. The formula is as Eq. 7.

$$SCP'(s) = \frac{\frac{SCP(s) - \mu}{3\sigma} + 1}{2} \tag{7}$$

Though Eq. 5, we evaluate the probability of a given character sequence being a valid word by combining its outer variance and inner cohesiveness. Basically, a valid Chinese word should occur in different environments (high BE) and the characters in the word should often occur together (high SCP). Take the previous "门把手 (doorknob)" example in the Branching Entropy part. We have shown that BE(门把手) is high, and from the corpus, we can see that character sequences "门把", "把手", "门把手" also have high co-occurrence which makes the overall P_{word}(门把手) high. The w_{BE} and w_{SCP} in Eq. 5 are the weights of $BE'(s)$ and $SCP'(s)$ in calculating $P_{validword}(s)$. Intuitively, we will set w_{BE} and w_{SCP} with the same value(i.e. $w_{BE} = 0.5$ and $w_{SCP} = 0.5$). However, from our study, we find that the BE value is more useful in Chinese new word detection in tweets. The reason might be that some of the "*Tweet-born*" new words are compound of several known words. Take the valid word "余额宝" (Yu-E Bao) "as an example, it is a compound of other two valid words "余额(balance)" and "宝(treasure)", so the denominator of SCP(余额宝) which contains *freq*(余额) × *freq*(宝) will become large and make the overall SCP(余额宝) relative small. In other words, SCP(余额宝) cannot efficiently identify "余额宝 (Yu-E Bao)" as an valid word in this case. Based on the above observation, we set w_{BE} slightly higher than w_{SCP}, which are 0.6 and 0.4 respectively.

Noise Filtering. Although a set of valid word candidates can be achieved by setting a threshold on $P_{word}(s)$, substrings of valid words exist as noise from our observation. For example, "国媒体(country media)" is an invalid segmentation, it is a substring of word/phrase such as "美国媒体(American media)" or "中国媒体(Chinese media)". However, P_{word}(国媒体) is not low. That is because on one hand, "国媒体(country media)" often appears together such that SCP(国媒体) is high. On the other hand, the preceding of "国媒体(country media)" can be "中[国](China)", "美[国](American)", "外[国](foreign countries)", etc. while the succeeding of "国媒体(country media)" can be "宣称(claim)", "报道(report)" which makes BE(国媒体) high as well. In other words, the character "国" is a component of many different words which makes the character sequences "国媒体" occurs in different environments even it is not a valid word. The basic idea of filtering this kind of noise is to consider word probability of the given character sequence and its overlapping strings [6]. Given character sequence $s = c_0 ... c_n$, the left overlapping string of s is $s_L = c_{-m} ... c_0 ... c_n$, the $c_{-m} ... c_{-1}$ is the m-character preceding sequence of s, the value of m is in the set $\{1, ..., n\}$ and denote the set of k-character preceding sequence as S_L. The left overlapping score of s is then calculated as Eq. 8.

$$OV_{left}(s) = \frac{\sum\limits_{s_L \in S_L(s)} I(s_L)}{|S_L(s)|} \tag{8}$$

Here $I(\cdot)$ is the indicator function defined as Eq. 9

$$I(s_L) = \begin{cases} 0 & P_{word}(s_L) \leq P_{word}(s) \\ 1 & P_{word}(s_L) > P_{word}(s) \end{cases} \tag{9}$$

The right overlapping score of s, $OV_{right}(s)$, can be calculated similarly. The overall overlapping score is defined as Eq. 10.

$$OV(s) = max\{OV_{left}(s), OV_{right}(s)\} \tag{10}$$

Character sequences with OV value larger than certain threshold are eliminated from the set of valid word candidates. A dictionary will serve as the knowledge base, any word in the dictionary will be consider as its $P_{word}(\cdot)$ is ∞. Then take the “国媒体” (country media) case as an example again, “美国(American)”, “中国(China)”, “美国媒体(American media)”, etc. are elements in S_L(国媒体) such that the left overlapping score of “国媒体” is high. That is because P_{word}(美国) $>$ P_{word}(国媒体), P_{word}(中国) $>$ P_{word}(国媒体), P_{word}(美国媒体) $>$ P_{word}(国媒体) and we can find many such cases using character sequences in S_L(国媒体) so the value OV(国媒体) is large which helps to identify “国媒体” as a wrong segmentation.

Overall Procedure of New Word Detection. Generally speaking, most of the Chinese words contain 2–4 characters, so we just consider character sequences whose length between two and four in the tweet corpus. We first select all the frequent character sequences, then calculate word probability of these character sequences bases on their SCP and BE score. Noises are filtered by the OV score of the character sequences afterward. Figure 1 shows an overview of the process of new word detection.

And the pseudo code of the procedure of new word detection is listed in Algorithm 1. Recall $Set_{tweet}(t)$ is the tweet corpus at time t, $Set_{tweet}(t_0)$ is the tweet corpus at time t_0 and let $Thres_{freq}$, $Thres_{wordprob}$ and $Thres_{OV}$ denotes the thresholds of character frequency, word probability score and overlapping score respectively.

5 New Word Tagging

The proposed idea of new word interpretation is automatic annotating a new word by tagging it with known words. Tagging is being extensively used in images (photos on facebook) [10] or articles annotation [11]. The objective of new word tagging is to shed light on the meaning of the word and facilitate users' better understanding. The objective of new word tagging is to shed light on the meaning of the word and facilitate users' better understanding. Our idea is to find out a list of known words that are most relevant to the given new word. Words in the following 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 people, organizations or microblog user's

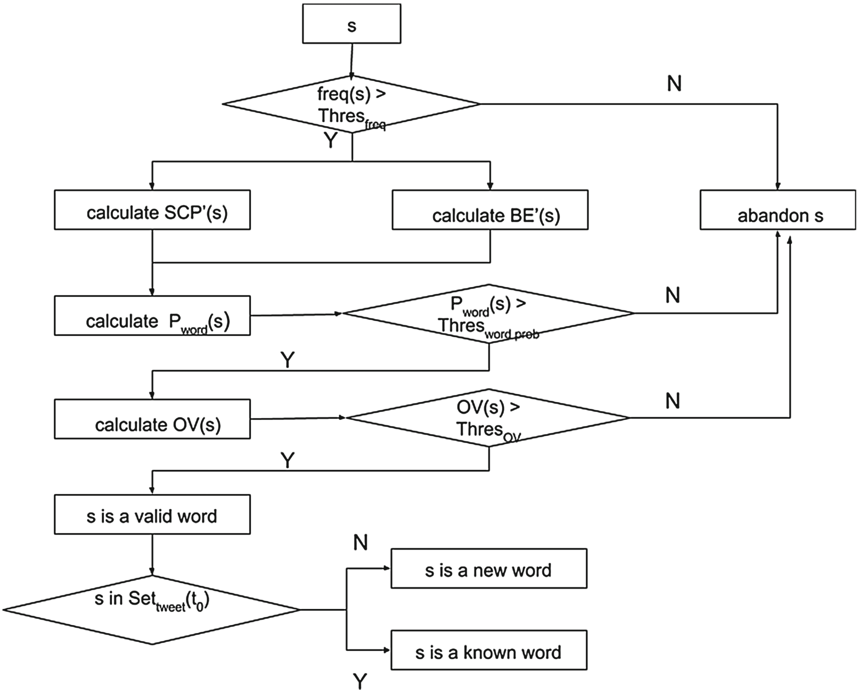

Fig. 1. Process of new word detection

comments which relate to the event. Normally, these words frequent co-occur with w_{new}. On the other hand, the second category, w_{new}'s synonyms, may not co-occur with w_{new} since the users often choose either w_{new} or its synonyms, but not both of them in the same tweet. For instance, "火星哥" (Mars brother) is a nickname of "华晨宇" (Hua Chenyu, the name of 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 for 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.

Given a new word w_{new}, we select the known words which are similar to w_{new} as its tag words. The similarity evaluation often falls into two subcategories, one is distance based and the other is angle based. In this paper, we propose two approaches, namely Context Distance and Context Cosine Similarity, to measure the similarity of the new words and known words from these two aspects. Given a new word, Context Distance finds tags of the new word by the distance between the known words and the new word while Context Cosine Similarity selects tag words based on the angle between the vectors of the known words and the given new word. The result shows Context Cosine Similarity can pick tag words more precisely compares to that of Context Distance.

Algorithm 1. New Word Detection

```
for all $s \in Set_{tweet}(t), 2 \leq |s| \leq 4$ do
    if $freq(s) \geq Thres_{freq}$ then
        Calculate $SCP(s)$ using Eq. 1
        Get the normalized $SCP(s)$, $SCP'(s)$, using Eq. 7
        Calculate $BE(s)$ using Eq. 4
        Get the normalized $BE(s)$, $BE'(s)$, using Eq. 6
        Get the word probability score $P_{word}(s)$ using Eq. 5
        if $P_{word}(s) \geq Thres_{wordprob}$ then
            Add $s$ to word candidate set $Set_{cand}$
        end if
    end if
end for
for all $s \in Set_{cand}$ do
    Calculate $OV(s)$ using Eq. 10
    if $OV(s) < Thres_{OV}$ then
        if $s \notin Set_{tweet}(t_0)$ then
            Add $s$ to $Set_{new}$
        else
            Add $s$ to $Set_{known}$
        end if
    end if
end for
```

5.1 Context Distance

From our study, the surrounding text of a word may shed light on its meaning. We could simply model the context of w_{known} as the set of words co-occurring with w_{new} in $Set_{tweet}(t)$. Let $doc(w_1, w_2)$ denotes the pseudo document made by concatenation of all tweets containing w_1 while w_1, w_2 are excluded from the document, we can get $doc(w_{new}, w_{known})$ and $doc(w_{known}, w_{new})$ using all the tweets containing w_{new} and w_{known} respectively. For example, the new word w_{new} is represented by a vector $v(w_{new}) = v_1, v_2, ..., v_k$, the ith element $v_i(w_{new})$ in $v(w_{new})$ indicates the relationship between w_{new} and the ith known word w_i. The value of $v_i(w_{new})$ is defined by Eq. 11. The length of $v(w_{new})$, denoted as k, equals to the size of the Set_{known}.

$$v_i(w_{new}) = \frac{tf(w_i, doc(w_{new}, w_i)) \times idf(w_i, D)}{|doc(w_{new}) \in D|} \tag{11}$$

The numerator is the term frequency - inverse document frequency (TF-IDF) weight of w_i which is a numerical statistic that is intended to reflect how important w_i is to $doc(w_{new}, w_i)$ in the tweet corpus D. Here we use $Set_{tweet}(t_0)$ as our tweet corpus and we use the occurrence of w_i as its term frequency (Eq. 12) and the proportion of tweets containing w_i in $doc(w_{new}, w_i)$ (before w_i is excluded from the document) is the inverse document frequency of w_i (Eq. 13).

The denominator is the number of tweets containing w_{new} among all the tweets.

$$tf(w_i, doc(w_{new}, w_{known})) = \frac{freq(w_i, doc(w_{new}, w_i))}{\sum\limits_{w \in doc(w_{new}, w_i)} freq(w, doc(w_{new}, w_i))} \tag{12}$$

$$idf(w_i, D) = \log \frac{|D|}{|doc(w_i) \in D|} \tag{13}$$

Where $doc(w_i)$ is all the tweets containing w_i.

Similarly, we can vectoring a known word w_{known} with $doc(w_{known}, w_{new})$. It is worthy noting that w_{new} and w_{known} are excluded from $doc(w_{new}, w_{known})$ and $doc(w_{known}, w_{new})$ because we assume if two words are semantically similar, their context should be similar even they co-occur with low frequency.

After using $v(w_{new})$ and $v(w_{known})$ to represent the new word and the known word as two vectors in high dimensional space, we can evaluate the similarity of w_{new} and w_{known} using their Context Distance. The proposed Context Distance is calculated by Euclidean Distance (Eq. 14) of the two words. The w_{new}, w_{known} are similar if the $dist(w_{new}, w_{known})$ is small. k equals to the size of the known word set.

$$dist(w_{new}, w_{known}) = \sqrt{\sum_{i=1}^{k} (v_i(w_{new}) - v_i(w_{known}))^2}. \tag{14}$$

The overall procedure of calculating Context Distance is listed in Algorithm 2.

5.2 Context Cosine Similarity

Different from Context Distance which evaluating the similarity of the new word and known the word by data point distance, Context Cosine Similarity define their similarity by the angle between the new word and known word vectors. The new word and the known word are similar if the angle between them is small.

Let $v''(w_{new})$ denotes the new word vector of Context Cosine Similarity, the ith element $v''_i(w_{new})$ in $v(w_{new})$ is defined by Eq. 15.

$$v_i(w_{new}) = tf(w_i, doc(w_{new}, w_i)) \tag{15}$$

The vector of known word w_{known}, $v''(w_{known})$, can be generated similarly. The angle between the two words is indicated by the cosine similarity of $v''(w_{new})$ and $v''(w_{known})$ (Eq. 16).

$$\begin{aligned} sim(w_{new}, w_{known}) &= \frac{v''(w_{new}) \cdot v''(w_{known})}{|v''(w_{new})||v''(w_{known})|} \\ &= \frac{\sum\limits_{i=1}^{k} v''_i(w_{new}) \times v''_i(w_{known})}{\sqrt{\sum\limits_{i=1}^{k} v''_i(w_{new})^2} \times \sqrt{\sum\limits_{i=1}^{k} v''_i(w_{known})^2}} \end{aligned} \tag{16}$$

$sim(w_{new}, w_{known})$ will be a value between 0 and 1, the higher the value is, the new word and the known word are more relevant.

Algorithm 2. Calculate Context Distance

```
Given a new word $w_{new}$ and a known word $w_{known}$
for all $T \in Set_{tweet}(t)$ do
    if $T$ contains $w_{new}$ then
        Add $T$ to $doc(w_{new})$
    end if
    if $T$ contains $w_{known}$ then
        Add $T$ to $doc(w_{known})$
    end if
end for
Get $doc(w_{new}, w_{known})$ by excluding all $w_{new}$ and $w_{known}$ from $doc(w_{new})$
Get $doc(w_{known}, w_{new})$ by excluding all $w_{known}$ and $w_{new}$ from $doc(w_{known})$
for all $w_i \in Set_{known}$ do
    Count $freq(w_i, doc(w_{new}, w_i))$
    Count $freq(w_i, doc(w_i, w_{new}))$
    for all $T \in Set_{tweet}(t)$ do
        if $T$ contains $w_i$ then
            Put $T$ into $doc(w_i)$
        end if
    end for
    Calculate $idf(w_i, Set_{tweet}(t))$ using Eq. 13)
    Calculate $v_i(w_{new})$, $v_i(w_{known})$ using Eq. 11
end for
Calculate $dist(w_{new}, w_{known})$ using Eq. 14
```

5.3 Choose Tag Words

Since the Context Distance and Context Cosine Similarity is relevant to the number of known words occur in $doc(w_{new}, w_{known})$, the average value and the standard variation is different for distinct new words. In other words, we cannot set a unified threshold for all the tag words of different new words directly. In this case, we perform a further max-min normalization on the top 20 relevant tag word of a specific new word. The max-min normalization formula is as Eq. 17.

$$dist'(i) = \frac{dist(i) - min_{dist}}{max_{dist} - min_{dist}} \tag{17}$$

After the max-min normalization, we can set a unified threshold $Thres_{CD}$ for all the new words in tag words selecting. The process of choosing tag words bases on Context Cosine Similarity is similar, but sorting the values in Set_{sim} by value descending then choosing the largest 20 values and w_{known} is considered as a tag word when the further normalized Context Cosine Similarity $sim'(w_{known})$ is larger than a threshold $Thres_{CCS}$.

The process of selecting tag words bases on Context Distance is listed as Algorithm 3.

The process of selecting tag words bases on Context Cosine Similarity is similar, but it is worth noticing we assuming w_{new} and w_{known} are relevant when the Context Distance $dist(w_{new}, w_{known})$ is small while assuming they are

Algorithm 3. Choose Tag Words Base on Context Distance

```
Given a new word $w_{new}$
for all $w_{known} \in Set_{known}$ do
    Using Eq. 14 to get $dist(w_{new}, w_{known})$
    Add $dist(w_{new}, w_{known})$ to $Set_{dist}$
end for
Sort $Set_{dist}$ by its value ascending and choose the top 20 values as $Set'_{dist}$
Select the max value in $Set'_{dist}$ as $max_{dist}$
Select the minimal value in $Set'_{dist}$ as $min_{dist}$
for all $dist \in Set'_{dist}$ do
    Using Eq. 17 to normalize the distance, $dist'(w_{new}, w_{known})$ is in range [0,1]
    if $dist'(w_{new}, w_{known}) \leq Thres_{CD}$ then
        $w_{known}$ is selected as a tag word of $w_{new}$
    end if
end for
```

relevant when the Context Cosine Similarity $sim(w_{new}, w_{known})$ is large, so we should sort the elements in the CCS value set Set_{sim} by value descending rather than ascending.

6 Experiment

6.1 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, $Set_{tweet}(t)$ 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 $Set_{tweet}(t_0), t_0 < t$ which serves as knowledge base. Hash tags, spam tweets and tweets only contains non-Chinese characters are filtered in the dataset. Table 2 shows the details of our dataset. And we store any character sequence with length between two and four in $Set_{tweet}(t_0)$ to serve as the known word set $Set_{word}(t_0)$ to ensure new words detected from $Set_{tweet}(t)$ has never appeared in $Set_{tweet}(t_0)$.

We perform cleaning on dataset used as $Set_{tweet}(t)$, where hash tags, spam tweets, tweets only contains non-Chinese characters are rejected. We store any character sequence with length between two and four in $Set_{tweet}(t_0)$ to serve as the known word set $Set_{word}(t_0)$ to ensure new words detected from $Set_{tweet}(t)$ has never appeared in $Set_{tweet}(t_0)$. Table 2 shows the details of our dataset. From Table 2, we can see that there are over 20 million tweets of $Set_{tweet}(t_0)$, around 50 times larger than the size of $Set_{tweet}(t)$, such that we assume $Set_{tweet}(t_0)$ is sufficient to server as our knowledge base. Moreover, even there is any word do not exist in $Set_{tweet}(t_0)$ but frequently appear in $Set_{tweet}(t)$, we assume the word becomes hot again for a new reason, this type of words also deserves our attention and needs to be annotated.

Table 2. 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	-

6.2 New Word Detection Result

In the new word detection experiment, we use ICTCLAS and Stanford Chinese-word-segmenter [21] to serve as our baselines. The training data used by ICTCLAS is Peking University dataset which contains 149,922 words while training 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 a new word. Non-Chinese character, emotion icon, punctuation, date, word containing stop words and some common words are excluded because they are not our target. 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, we just evaluating the word probability of the top 5% frequent character sequence and set the threshold $Thres_{freq}$ to 15, words appearing less than $Thres_{freq}$ will be ignored. Figure 2 shows the character sequence amount of different frequency. We can see that most of the character sequences occur less than 5 times. However, from Fig. 3 we can notice that the low-frequency character sequences often not valid words. Take the character sequences occur less than 5 times as an example, only 5% of them are valid words.

Generally speaking, Chinese new words can be divided into several categories [20] (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 Eq. 18

$$Precision = \frac{\#\text{ of valid new words}}{\#\text{ of total new words detected}} \tag{18}$$

The threshold of overlapping score, $Thres_{OV}$, is set to 0.7, same as that in [6]. We set the threshold of the word probability, $Thres_{wordprob}$, to 0.3 bases on the following observation: the word probability of all the character sequences mainly fall into the range [0, 1] and there are around 30% of the character sequences whose frequency larger than $Thres_{freq}$ are an invalid word.

The experiment results are listed in Table 3.

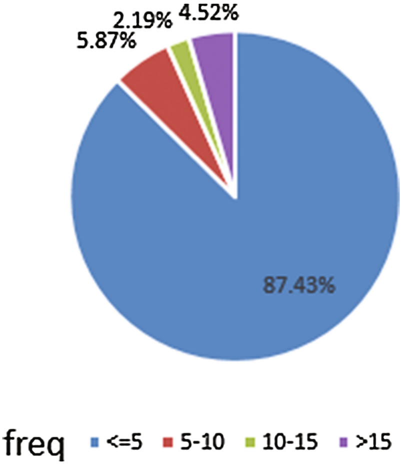

Fig. 2. Character sequence proportion of different frequency

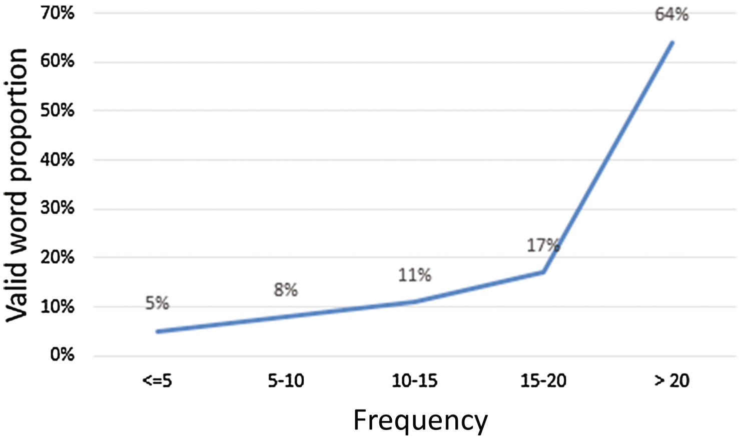

Fig. 3. Valid word proportion of different frequency

The 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 "*Wechat Wo card*". For example, for the sentence "余额宝将让中国银行倒闭" (*Yu'E Bao* will make Chinese banks bankrupt), "余额宝" (*Yu'E Bao*) is a totally new product while "余额" is a

Table 3. 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%

known word which probably exists in the previous corpus, the segmentation result constructed by ICTCLAS is 余额/宝/ 将/让/ 中国/ 银行/倒闭 which separates "余额/宝/ 将/让/ 中国/ 银行/倒闭" (Yu'E) and "宝" (Bao) as two different words. To summarize, our method is more effective than the two baselines in term of recall and precision as a whole.

6.3 Tagging Result Bases on Context Distance

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:

$$Precision_{tag}(w_{new}) = \frac{\#\text{ of tag words hits in } w_{new}\text{'s Baidu Entry}}{\#\text{ of } w_{new}\text{'s tag words}} \tag{19}$$

Known words whose normalized Context Distance lower than a threshold $Thres_{CD}$ are selected as the tag of the new word. 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. We have tried different $Thres_{CD}$ and compare the tagging accuracy and the number of tag words is selected. The result is in Table 4.

Table 4. Word tagging result of different $Thres_{CD}$

Threshold	Average tag count	Average precision
0.25	2.2	0.65
0.5	4.4	0.55
0.75	8.3	0.52
1	20	0.44

From Table 4 we can see that the number of selected tag words decreases while the tagging precision increase when $Thres_{CD}$ decline. This is in consistent with tag words which have smaller Context Distance with the new word is more likely be the right tag. Generally, we seek for high tagging accuracy while need enough number of tags to have a detailed interpretation of the given new word. We can achieve the highest precision when we set $Thres_{CD} = 0.25$, however, only 2.2 known words are selected as tag words of the given new word in this case which make the interpretation indistinctly.

According to Table 4, we set $Thres_{CD} = 0.75$ to get a good balance between the tagging accuracy and the number of tag words. Some tagging result are listed as below.

- **Yu'E Bao** (Novelty. Yu'E Bao is a money-market fund promoted by Alipay)
  Tags: Suning Online Market (An online shop which can using Yu'E Bao to pay), fund, money management
- **Wechat Wo card** (Novelty. A SIM card released by Tencent and China Unicom. People using Wo card can have some special rights in Wechat)
  Tags: special right, China Unicom, Wechat, Tencent
- **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: idol, China, sound, succeed, summer, young, end, perfect
- **Wu Mulan** (Name Entity. A player in the TV program "The voice of China". An infertility patients claimed she get pregnant after listen to Wu's song since the music made she relax)
  Tags: voice, China, song, enjoy, music, pregnant, view, sprit, child, support, talk, strong, sing
- **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: eat frequently, growth, mushroom, virus, immunity
- **Burst Bar event** (Abbreviation of hot topic. Pan Mengyin, a fan of Korean star G-Dragon, spread some inappropriate remarks about football stars on Internet which makes fans of the football stars get angry and attacked G-Dragon's Baidu Bar)
  Tags: G-Dragon, hope, friend, whole life, Pan Mengyin, Internet, strong, birthday, strength

Moreover, we have further compare the number of tags and the tagging precision of different word categories. The result is in Table 5.

6.4 Tagging Result Bases on Context Cosine Similarity

Known words whose normalized Context Cosine Similarity higher than a threshold $Thres_{CCS}$ are selected as the tag of the new word. The tagging accuracy and the number of tag words of different $Thres_{CCS}$ is shown in Table 6.

Table 5. Word tagging result of different categories

Category	New word count	Average tag count	Average precision
Name entity	9	9.3	0.45
Glossary	1	5.0	0.00
Novelty	4	7.5	0.62
Abbreviation of hot topic	6	7.8	0.66

Table 6. Word tagging result of different thresholds

Threshold	Average tag count	Average precision
0	19.6	0.56
0.25	9.1	0.71
0.5	5.5	0.79
0.75	3	0.825

Table 6 shows the number of selected tag words decreases while the tagging precision increase when the threshold of Context Cosine Similarity arise. This indicates the tag words which have higher context cosine similarity with the new word is more likely be the right tag of the new word. In our experiment, we set the threshold $Thres_{CCS} = 0.5$ such that we can get enough tag words while achieving a relatively high precision. Some tagging result are listed as below.

– **Yu'E Bao** (Novelty. Yu'E Bao is a money-market fund promoted by Alipay)
  Tags: currency, Internet, finance, fund, money management, supply-chain
– **Wechat Wo card** (Novelty. A SIM card released by Tencent and China Unicom. People using Wo card can have some special rights in Wechat)
  Tags: special right, China Unicom, Tencent, network traffic
– **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
– **Wu Mulan** (Name Entity. A player in the TV program "The voice of China". An infertility patients claimed she get pregnant after listen to Wu's song since the music made she relax)
  Tags: Mushroom Brothers, Liu Zichen, Yu Junyi, Ta Siken, Ni peng, supervisor (The first five tags are other players in the same show, the last tag is a role in the show)
– **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, immunity, vegetables, growth

- **Burst Bar event** (Abbreviation of hot topic. Pan Mengyin, a fan of Korean star G-Dragon, spread some inappropriate remarks about football stars on Internet which makes fans of the football stars get angry and attacked G-Dragon's Baidu Bar)
  Tags: Pan Mengyin, G-Dragon, Korean star, stupid

We also further compared the number of tags and tagging precisions of different word categories in Table 7).

Table 7. Word tagging result of different categories

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

6.5 Tagging Result Discussion

Comparing the tagging result bases on Context Distance and that of Context Cosine Similarity, we found Context Cosine Similarity performs better among the two approaches. By comparing Tables 4 and 6, we can notice that for a similar number of selected tag words, Context Cosine Similarity often gets a higher precision. For example, in Table 4, when $Thres_{CD} = 1$, there are 20 tag words selected for each new word on average, the precision of the tag word selection is 0.44. That compares to $Thres_{CCS} = 0$ in Table 6, there are 19.1 tag words selected for each new word on average, the precision of the tag word selection achieves 0.56. This phenomena might because in Context Distance calculation, although tf-idf has been utilized to reduce the impact of the imbalance of word frequency, some common words are still selected as a tag word of a new word even they are not highly relevant (e.g. "friend" and "birthday" are selected as the tag words of "*Pan Mengyin*"). Context Cosine Similarity overpass the word frequency imbalance problem by using data distribution to define the similarity value.

By analyzing the word tagging results of different categories (Tables 5 and 7), we can see an interesting thing that comparing to name entity and abbreviation of hot topic, novelty have fewer number of tag words while achieves much higher precision. Both of Context Distance and Context Cosine Similarity failed in tagging the glossary *Ergosterol* precisely because a lot of tweets talking *Ergosterol* are a kind of advertisement. Moreover, even some related words are selected as a tag word of *Ergosterol*, the online encyclopedia Baidu Entry using more technical terms to explain *ergosterol*. For example, mushroom is a tag word of *Ergosterol* bases on Context Distance and mushroom contains a lot of Ergosterol so these two words are relevant, but in Baidu Entry, it used the term *Fungus* instead of mushroom.

7 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. The proposed framework detect new words based on statistics information such as sequence frequency, the specific wording do not serve as features in this case, so it could be easily applied to other Kanji based languages (e.g. Japanese and Korean).

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. Peng, F., Feng, F., McCallum, A.: Chinese segmentation and new word detection using conditional random fields. In: Proceedings of the 20th International Conference on Computational Linguistics, p. 562. Association for Computational Linguistics (2004)
2. Finin, T., et al.: Annotating named entities in Twitter data with crowdsourcing. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics (2010)
3. Ritter, A., Clark, S., Etzioni, O.: Named entity recognition in tweets: an experimental study. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2011)
4. Gattani, A., et al.: Entity extraction, linking, classification, and tagging for social media: a wikipedia-based approach. Proc. VLDB Endow. **6**(11), 1126–1137 (2013)
5. Sun, X., Wang, H., Li, W.: Fast online training with frequency-adaptive learning rates for Chinese word segmentation and new word detection. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers, vol. 1. Association for Computational Linguistics (2012)
6. Ye, Y., Qingyao, W., Li, Y., Chow, K.P., Hui, L.C.K., Kwong, L.C.: Unknown Chinese word extraction based on variety of overlapping strings. Inf. Process. Manag. **49**(2), 497–512 (2013)
7. Zhao, H., Kit, C.: Exploiting unlabeled text with different unsupervised segmentation criteria for Chinese word segmentation. Res. Comput. Sci. **33**, 93–104 (2008)
8. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **30**(1), 3–26 (2007)

9. Zhao, H., Kit, C.: Unsupervised segmentation helps supervised learning of character tagging for word segmentation and named entity recognition. In: IJCNLP, pp. 106–111 (2008)
10. Zhou, N., et al.: A hybrid probabilistic model for unified collaborative and content-based image tagging. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1281–1294 (2011)
11. Kim, H.-N., et al.: Collaborative filtering based on collaborative tagging for enhancing the quality of recommendation. Electron. Commer. Res. Appl. **9**(1), 73–83 (2010)
12. Luo, S., Sun, M.: Two-character Chinese word extraction based on hybrid of internal and contextual measures. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, vol. 17. Association for Computational Linguistics (2003)
13. Jin, Z., Tanaka-Ishii, K.: Unsupervised segmentation of Chinese text by use of branching entropy. In: Proceedings of the COLING/ACL on Main Conference Poster Sessions, Association for Computational Linguistics (2006)
14. Wang, L., et al.: CRFs-based Chinese word segmentation for micro-blog with small-scale data. In: Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language (2012)
15. Zhang, K., Sun, M., Zhou, C.: Word segmentation on Chinese mirco-blog data with a linear-time incremental model. In: Proceedings of the Second CIPS-SIGHAN Joint Conference on Chinese Language Processing, Tianjin (2012)
16. Zhang, H.-P., et al.: HHMM-based Chinese lexical analyzer ICTCLAS. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, vol. 17. Association for Computational Linguistics (2003)
17. Dumais, S.T.: Latent semantic analysis. Annu. Rev. Inf. Sci. Technol. **38**(1), 188–230 (2004)
18. Aksoy, S., Haralick, R.M.: Feature normalization and likelihood-based similarity measures for image retrieval. Pattern Recognit. Lett. **22**(5), 563–582 (2001)
19. Kityz, C., Wilksz, Y.: Unsupervised learning of word boundary with description length gain. In: Proceedings of the CoNLL99 ACL Workshop. Association for Computational Linguistics, Bergen (1999)
20. Gang, Z., et al.: Chinese new words detection in internet. Chin. Inf. Technol. **18**(6), 1–9 (2004)
21. Tseng, H., et al.: A conditional random field word segmenter for sighan bakeoff 2005. In: Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, Jeju Island, vol. 171 (2005)
22. Zheng, X., Chen, H., Xu, T.: Deep learning for Chinese word segmentation and POS tagging. In: EMNLP (2013)
23. Wallach, H.M.: Conditional random fields: an introduction (2004)