New Word Detection for Sentiment Analysis

Minlie Huang, Borui Ye*, Yichen Wang, Haiqiang Chen, Junjun Cheng**, Xiaoyan Zhu**
State Key Lab. of Intelligent Technology and Systems, National Lab. for Information Science and Technology, Dept. of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China
*Dept. of Communication Engineering, Beijing University of Posts and Telecommunications
**China Information Technology Security Evaluation Center
aihuang@tsinghua.edu.cn

Abstract

Automatic extraction of new words is an indispensable precursor to many NLP tasks such as Chinese word segmentation, named entity extraction, and sentiment analysis. This paper aims at extracting *new sentiment words* from large-scale user-generated content. We propose a fully unsupervised, purely data-driven framework for this purpose. We design statistical measures respectively to quantify the utility of a lexical pattern and to measure the possibility of a word being a new word. The method is almost free of linguistic resources (except POS tags), and requires no elaborated linguistic rules. We also demonstrate how new sentiment word will benefit sentiment analysis. Experiment results demonstrate the effectiveness of the proposed method.

1 Introduction

New words on the Internet have been emerging all the time, particularly in user-generated content. Users like to update and share their information on social websites with their own language styles, among which new political/social/cultural words are constantly used.

However, such new words have made many natural language processing tasks more challenging. Automatic extraction of new words is indispensable to many tasks such as Chinese word segmentation, machine translation, named entity extraction, question answering, and sentiment analysis. New word detection is one of the most critical issues in Chinese word segmentation. Recent studies (Sproat and Emerson, 2003) (Chen, 2003) have shown that more than 60% of word segmentation errors result from new words. Statistics show that more than 1000 new Chinese words appear every year (Thesaurus Research Center, 2003). These words are mostly domain-specific technical terms and time-sensitive political/social /cultural terms. Most of them are not yet correctly recognized by the segmentation algorithm, and remain as out of vocabulary (OOV) words.

New word detection is also important for sentiment analysis such as opinionated phrase extraction and polarity classification. A sentiment phrase with complete meaning should have a correct boundary, however, characters in a new word may be broken up. For example, in a sentence " 表演/n 非常/adv 给/v 力/n（artists' performance is very impressive）" the two Chinese characters "给/v 力/n(cool; powerful)" should always be extracted together. In polarity classification, new words can be informative features for classification models. In the previous example, "给力(cool; powerful)" is a strong feature for classification models while each single character is not. Adding new words as feature in classification models will improve the performance of polarity classification, as demonstrated later in this paper.

This paper aims to detect new word for sentiment analysis. We are particulary interested in extracting *new sentiment word* that can express opinions or sentiment, which is of high value towards sentiment analysis. *New sentiment word*, as exemplified in Table 1, is a sub-class of multi-word expressions which is a sequence of neighboring words *"whose exact and unambiguous meaning or connotation cannot be derived from the meaning or connotation of its components"* (Choueka, 1988). Such new words cannot be directly identified using grammatical rules, which poses a major challenge to automatic analysis. Moreover, existing lexical resources never have adequate and timely coverage since new words appear constantly. People thus resort to statistical methods such as Pointwise Mutual Information (Church and Hanks, 1990), Symmetrical Conditional Probability

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 531–541,
Baltimore, Maryland, USA, June 23-25 2014.

(da Silva and Lopes, 1999), Mutual Expectation (Dias et al., 2000), Enhanced Mutual Information (Zhang et al., 2009), and Multi-word Expression Distance (Bu et al., 2010).

New word	English Translation	Polarity
口爱	lovely	positive
杯具	tragic/tragedy	negative
给力	very cool; powerful	positive
坑爹	reverse one's expectation	negative

Table 1: Examples of new sentiment word.

Our central idea for new sentiment word detection is as follows: Starting from very few seed words (for example, just one seed word), we can extract lexical patterns that have strong statistical association with the seed words; the extracted lexical patterns can be further used in finding more new words, and the most probable new words can be added into the seed word set for the next iteration; and the process can be run iteratively until a stop condition is met. The key issues are to measure the utility of a pattern and to quantify the possibility of a word being a new word. The main contributions of this paper are summarized as follows:

- We propose a novel framework for new word detection from large-scale user-generated data. This framework is fully unsupervised and purely data-driven, and requires very lightweight linguistic resources (i.e., only POS tags).
- We design statistical measures to quantify the utility of a pattern and to quantify the possibility of a word being a new word, respectively. No elaborated linguistic rules are needed to filter undesirable results. This feature may enable our approach to be portable to other languages.
- We investigate the problem of polarity prediction of new sentiment word and demonstrate that inclusion of new sentiment word benefits sentiment classification tasks.

The rest of the paper is structured as follows: we will introduce related work in the next section. We will describe the proposed method in Section 3, including definitions, the overview of the algorithm, and the statistical measures for addressing the two key issues. We then present the experiments in Section 4. Finally, the work is summarized in Section 5.

2 Related Work

New word detection has been usually interweaved with word segmentation, particularly in Chinese NLP. In these works, new word detection is considered as an integral part of segmentation, where new words are identified as the most probable segments inferred by the probabilistic models; and the detected new word can be further used to improve word segmentation. Typical models include conditional random fields proposed by (Peng et al., 2004), and a joint model trained with adaptive online gradient descent based on feature frequency information (Sun et al., 2012).

Another line is to treat new word detection as a separate task, usually preceded by part-of-speech tagging. The first genre of such studies is to leverage complex linguistic rules or knowledge. For example, Justeson and Katz (1995) extracted technical terminologies from documents using a regular expression. Argamon et al. (1998) segmented the POS sequence of a multi-word into small POS tiles, counted tile frequency in the new word and non-new-word on the training set respectively, and detected new words using these counts. Chen and Ma (2002) employed morphological and statistical rules to extract Chinese new word. The second genre of the studies is to treat new word detection as a classification problem. Zhou (2005) proposed a discriminative Markov Model to detect new words by chunking one or more separated words. In (Li et al., 2005), new word detection was viewed as a binary classification problem. However, these supervised models requires not only heavy engineering of linguistic features, but also expensive annotation of training data.

User behavior data has recently been explored for finding new words. Zheng et al. (2009) explored user typing behaviors in Sogou Chinese Pinyin input method to detect new words. Zhang et al. (2010) proposed to use dynamic time warping to detect new words from query logs. However, both of the work are limited due to the public unavailability of expensive commercial resources.

Statistical methods for new word detection have been extensively studied, and in some sense exhibit advantages over linguistics-based methods. In this setting, new word detection is mostly

known as multi-word expression extraction. To measure multi-word association, the first model is Pointwise Mutual Information (PMI) (Church and Hanks, 1990). Since then, a variety of statistical methods have been proposed to measure bi-gram association, such as Log-likelihood (Dunning, 1993) and Symmetrical Conditional Probability (SCP) (da Silva and Lopes, 1999). Among all the 84 bi-gram association measures, PMI has been reported to be the best one in Czech data (Pecina, 2005). In order to measure arbitrary n-grams, most common strategies are to separate n-gram into two parts X and Y so that existing bi-gram methods can be used (da Silva and Lopes, 1999; Dias et al., 2000; Schone and Jurafsky, 2001). Zhang et al. (2009) proposed Enhanced Mutual Information (EMI) which measures the cohesion of n-gram by the frequency of itself and the frequency of each single word. Based on the information distance theory, Bu et al. (2010) proposed multi-word expression distance (MED) and the normalized version, and reported superior performance to EMI, SCP, and other measures.

3 Methodology

3.1 Definitions

Definition 3.1 (**Adverbial word**). Words that are used mainly to modify a verb or an adjective, such as "太(too)", "非常(very)", "十分(very)", and "特别(specially)".

Definition 3.2 (**Auxiliary word**). Words that are auxiliaries, model particles, or punctuation marks. In Chinese, such words are like "着,了,啦,的,啊", and punctuation marks include "，。！？；： " and so on.

Definition 3.3 (**Lexical Pattern**). A lexical pattern is a triplet $< AD, *, AU >$, where AD is an adverbial word, the wildcard $*$ means an arbitrary number of words [1], and AU denotes an auxiliary word.

Table 2 gives some examples of lexical patterns. In order to obtain lexical patterns, we can define regular expressions with POS tags [2] and apply the regular expressions on POS tagged texts. Since the tags of adverbial and auxiliary words are relatively static and can be easily identified, such a method can safely obtain lexical patterns.

Pattern	Frequency
<"都",*,"了">	562,057
<"都",*,"的">	387,649
<"太",*,"了">	380,470
<"不",*,"， ">	369,702

Table 2: Examples of lexical pattern. The frequency is counted on 237,108,977 Weibo posts.

3.2 The Algorithm Overview

The algorithm works as follows: starting from very few seed words (for example, a word in Table 1), the algorithm can find lexical patterns that have strong statistical association with the seed words in which the likelihood ratio test (LRT) is used to quantify the degree of association. Subsequently, the extracted lexical patterns can be further used in finding more new words. We design several measures to quantify the possibility of a candidate word being a new word, and the top-ranked words will be added into the seed word set for the next iteration. The process can be run iteratively until a stop condition is met. Note that we do not augment the pattern set ($\mathcal{P}$) at each iteration, instead, we keep a fixed small number of patterns during iteration because this strategy produces optimal results.

From linguistic perspectives, new sentiment words are commonly modified by adverbial words and thus can be extracted by lexical patterns. This is the reason why the algorithm will work. Our algorithm is in spirit to double propagation (Qiu et al., 2011), however, the differences are apparent in that: firstly, we use very lightweight linguistic information (except POS tags); secondly, our major contributions are to propose statistical measures to address the following key issues: first, to measure the utility of lexical patterns; second, to measure the possibility of a candidate word being a new word.

3.3 Measuring the Utility of a Pattern

The first key issue is to quantify the utility of a pattern at each iteration. This can be measured by the association of a pattern to the current word set used in the algorithm. The likelihood ratio tests (Dunning, 1993) is used for this purpose. This association model has also been used to model association between opinion target words by (Hai et

[1] We set the number to 3 words in this work considering computation costs.

[2] Such expressions are very simple and easy to write because we only need to consider POS tags of adverbial and auxiliary word.

Algorithm 1: New word detection algorithm

Input:
$\mathcal{D}$: a large set of POS tagged posts
$\mathcal{W}_s$: a set of seed words
k_p: the number of patterns chosen at each iteration
k_c: the number of patterns in the candidate pattern set
k_w: the number of words added at each iteration
K: the number of words returned
Output: A list of ranked new words $\mathcal{W}$

Obtain all lexical patterns using regular expressions on $\mathcal{D}$;
Count the frequency of each lexical pattern and extract words matched by each pattern ;
Obtain top k_c frequent patterns as candidate pattern set $\mathcal{P}_c$ and top 5,000 frequent words as candidate word set $\mathcal{W}_c$;
$\mathcal{P} = \Phi$; $\mathcal{W}$=$\mathcal{W}_s$; t = 0 ;
for $|\mathcal{W}| < K$ **do**
    Use $\mathcal{W}$ to score each pattern in $\mathcal{P}_c$ with $U(p)$;
    $\mathcal{P} = \{top\ k_p\ patterns\}$;
    Use $\mathcal{P}$ to extract new words and if the words are in $\mathcal{W}_c$, score them with $F(w)$;
    $\mathcal{W} = \mathcal{W} \bigcup \{top\ k_w\ words\}$;
    $\mathcal{W}_c$ = $\mathcal{W}_c$ - $\mathcal{W}$;
Sort words in $\mathcal{W}$ with $F(w)$;
Output the ranked list of words in $\mathcal{W}$;

al., 2012).

The LRT is well known for not relying critically on the assumption of normality, instead, it uses the asymptotic assumption of the generalized likelihood ratio. In practice, the use of likelihood ratios tends to result in significant improvements in text-analysis performance.

In our problem, LRT computes a contingency table of a pattern p and a word w, derived from the corpus statistics, as given in Table 3, where $k_1(w, p)$ is the number of documents that w matches pattern p, $k_2(w, \bar{p})$ is the number of documents that w occurs while p does not, $k_3(\bar{w}, p)$ is the number of documents that p occurs while w does not, and $k_4(\bar{w}, \bar{p})$ is the number of documents containing neither p nor w.

Statistics	p	$\bar{p}$
w	$k_1(w, p)$	$k_2(w, \bar{p})$
$\bar{w}$	$k_3(\bar{w}, p)$	$k_4(\bar{w}, \bar{p})$

Table 3: Contingency table for likelihood ratio test (LRT).

Based on the statistics shown in Table 3, the likelihood ratio tests (LRT) model captures the statistical association between a pattern p and a word w by employing the following formula:

$$LRT(p, w) = log \frac{L(\rho_1, k_1, n_1) * L(\rho_2, k_2, n_2)}{L(\rho, k_1, n_1) * L(\rho, k_2, n_2)} \quad (1)$$

where:
$L(\rho, k, n) = \rho^k * (1 - \rho)^{n-k}$; $n_1 = k_1 + k_3$; $n_2 = k_2 + k_4$; $\rho_1 = k_1/n_1$; $\rho_2 = k_2/n_2$; $\rho = (k_1 + k_2)/(n_1 + n_2)$.

Thus, the utility of a pattern can be measured as follows:

$$U(p) = \sum_{w_i \in \mathcal{W}} LRT(p, w_i) \quad (2)$$

where $\mathcal{W}$ is the current word set used in the algorithm (see Algorithm 1).

3.4 Measuring the Possibility of Being New Words

Another key issue in the proposed algorithm is to quantify the possibility of a candidate word being a new word. We consider several factors for this purpose.

3.4.1 Likelihood Ratio Test

Very similar to the pattern utility measure, LRT can also be used to measure the association of a candidate word to a given pattern set, as follows:

$$LRT(w) = \sum_{p_i \in \mathcal{P}} LRT(w, p_i) \quad (3)$$

where $\mathcal{P}$ is the current pattern set used in the algorithm (see Algorithm 1), and p_i is a lexical pattern.

This measure only quantifies the association of a candidate word to the given pattern set. It tells nothing about the possibility of a word being a new word, however, a new *sentiment* word, should have close association with the lexical patterns. This has linguistic interpretations because new sentiment words are commonly modified by adverbial words and thus should have close association with lexical patterns. This measure is proved to be an influential factor by our experiments in Section 4.3.

3.4.2 Left Pattern Entropy

If a candidate word is a new word, it will be more commonly used with diversified lexical patterns since the non-compositionality of new word means that the word can be used in many different linguistic scenarios. This can be measured by information entropy, as follows:

$$LPE(w) = -\sum_{l_i \in L(\mathcal{P}_c, w)} \frac{c(l_i, w)}{N(w)} * log \frac{c(l_i, w)}{N(w)} \quad (4)$$

where $L(\mathcal{P}_c, w)$ is the set of left word of all patterns by which word w can be matched in $\mathcal{P}_c$, $c(l_i, w)$ is the count that word w can be matched by patterns whose left word is l_i, and $N(w)$ is the count that word w can be matched by the patterns in $\mathcal{P}_c$. Note that we use $\mathcal{P}_c$, instead of $\mathcal{P}$, because the latter set is very small while computing entropy needs a large number of patterns. Tuning the size of $\mathcal{P}_c$ will be further discussed in Section 4.4.

3.4.3 New Word Probability

Some words occur very frequently and can be widely matched by lexical patterns, but they are not new words. For example, " 爱吃(love to eat)" and " 爱说(love to talk)" can be matched by many lexical patterns, however, they are not new words due to the lack of non-compositionality. In such words, each single character has high probability to be a word. Thus, we design the following measure to favor this observation.

$$NWP(w) = \prod_{i=1}^{n} \frac{p(w_i)}{1 - p(w_i)} \quad (5)$$

where $w = w_1 w_2 \dots w_n$, each w_i is a single character, and $p(w_i)$ is the probability of the character w_i being a word, as computed as follows:

$$p(w_i) = \frac{all(w_i) - s(w_i)}{all(w_i)}$$

where $all(w_i)$ is the total frequency of w_i, and $s(w_i)$ is the frequency of w_i being a single character word. Obviously, in order to obtain the value of $s(w_i)$, some particular Chinese word segmentation tool is required. In this work, we resort to ICTCLAS (Zhang et al., 2003), a widely used tool in the literature.

3.4.4 Non-compositionality Measures

New words are usually multi-word expressions, where a variety of statistical measures have been proposed to detect multi-word expressions. Thus, such measures can be naturally incorporated into our algorithm.

The first measure is enhanced mutual information (EMI) (Zhang et al., 2009):

$$EMI(w) = log_2 \frac{F/N}{\prod_{i=1}^{n} \frac{F_i - F}{N}} \quad (6)$$

where F is the number of posts in which a multi-word expression $w = w_1 w_2 \dots w_n$ occurs, F_i is the number of posts where w_i occurs, and N is the total number of posts. The key idea of EMI is to measure word pair' s dependency as the ratio of its probability of being a multi-word to its probability of not being a multi-word. The larger the value, the more possible the expression will be a multi-word expression.

The second measure we take into account is normalized multi-word expression distance (Bu et al., 2010), which has been proposed to measure the non-compositionality of multi-word expressions.

$$NMED(w) = \frac{log|\mu(w)| - log|\phi(w)|}{logN - log|\phi(w)|} \quad (7)$$

where $\mu(w)$ is the set of documents in which all single words in $w = w_1 w_2 \dots w_n$ co-occur, $\phi(w)$ is the set of documents in which word w occurs as a whole, and N is the total number of documents. Different from EMI, this measure is a strict distance metric, meaning that a smaller value indicates a larger possibility of being a multi-word expression. As can be seen from the formula, the key idea of this metric is to compute the ratio of the co-occurrence of all words in a multi-word expressions to the occurrence of the whole expression.

3.4.5 Configurations to Combine Various Factors

Taking into account the aforementioned factors, we have different settings to score a new word, as follows:

$$F_{LRT}(w) = LRT(w) \quad (8)$$

$$F_{LPE}(w) = LRT(w) * LPE(w) \quad (9)$$

$$F_{NWP}(w) = LRT(w) * LPE(w) * NWP(w) \quad (10)$$

$$F_{EMI}(w) = LRT(w) * LPE(w) * EMI(w) \quad (11)$$

$$F_{NMED}(w) = \frac{LRT(w) * LPE(w)}{NMED(w)} \quad (12)$$

4 Experiment

In this section, we will conduct the following experiments: first, we will compare our method to several baselines, and perform parameter tuning with extensive experiments; second, we will classify polarity of new sentiment words using two methods; third, we will demonstrate how new sentiment words will benefit sentiment classification.

4.1 Data Preparation

We crawled 237,108,977 Weibo posts from http://www.weibo.com, the largest social website in China. These posts range from January of 2011 to December of 2012. The posts were then part-of-speech tagged using a Chinese word segmentation tool named ICTCLAS (Zhang et al., 2003).

Then, we asked two annotators to label the top 5,000 frequent words that were extracted by lexical patterns as described in Algorithm 1. The annotators were requested to judge whether a candidate word is a new word, and also to judge the polarity of a new word (positive, negative, and neutral). If there is a disagreement on either of the two tasks, discussions are required to make the final decision. The annotation led to 323 new words, among which there are 116 positive words, 112 negative words, and 95 neutral words[3].

4.2 Evaluation Metric

As our algorithm outputs a ranked list of words, we adapt average precision to evaluate the performance of new sentiment word detection. The metric is computed as follows:

$$AP(K) = \frac{\sum_{k=1}^{K} P(k) * rel(k)}{\sum_{k=1}^{K} rel(k)}$$

where $P(k)$ is the precision at cut-off k, $rel(k)$ is 1 if the word at position k is a new word and 0 otherwise, and K is the number of words in the ranked list. A perfect list (all top K items are correct) has an AP value of 1.0.

4.3 Evaluation of Different Measures and Comparison to Baselines

First, we assess the influence of likelihood ratio test, which measures the association of a word to the pattern set. As can be seen from Table 4, the association model (LRT) remarkably boosts the performance of new word detection, indicating LRT is a key factor for new sentiment word extraction. From linguistic perspectives, new sentiment words are commonly modified by adverbial words and thus should have close association with lexical patterns.

Second, we compare different settings of our method to two baselines. The first one is enhanced mutual information (EMI) where we set $F(w) = EMI(w)$ (Zhang et al., 2009) and the second baseline is normalized multi-word expression distance (NMED) (Bu et al., 2010) where we set $F(w) = NMED(w)$. The results are shown in Figure 1. As can be seen, all the proposed measures outperform the two baselines (EMI and $NMED$) remarkably and consistently. The setting of F_{NMED} produces the best performance. Adding $NMED$ or EMI leads to remarkable improvements because of their capability of measuring non-compositionality of new words. Only using LRT can obtain a fairly good results when K is small, however, the performance drops sharply because it's unable to measure non-compositionality. Comparison between $LRT + LPE$ (or $LRT + LPE + NWP$) and LRT shows that inclusion of left pattern entropy also boosts the performance apparently. However, the new word probability (NWP) has only marginal contribution to improvement.

In the above experiments, we set $k_p = 5$ (the number of patterns chosen at each iteration) and $k_w = 10$ (the number of words added at each iteration), which is the optimal setting and will be discussed in the next subsection. And only one seed word "坑爹(reverse one's expectation)" is used.

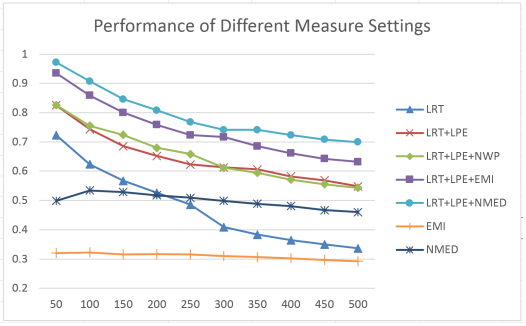

Figure 1: Comparative results of different measure settings. X-axis is the number of words returned (K), and Y-axis is average precision ($AP(K)$).

[3] All the resources are available upon request.

$top\ K\ words \Rightarrow$	100	200	300	400	500
LPE	0.366	0.324	0.286	0.270	0.259
LRT+LPE	**0.743**	**0.652**	**0.613**	**0.582**	**0.548**
LPE+NWP	0.467	0.400	0.350	0.330	0.320
LRT+LPE+NWP	**0.755**	**0.680**	**0.612**	**0.571**	**0.543**
LPE+EMI	0.608	0.551	0.519	0.486	0.467
LRT+LPE+EMI	**0.859**	**0.759**	**0.717**	**0.662**	**0.632**
LPE+NMED	0.749	0.690	0.641	0.612	0.576
LRT+LPE+NMED	**0.907**	**0.808**	**0.741**	**0.723**	**0.699**

Table 4: Results with vs. without likelihood ratio test (LRT).

4.4 Parameter Tuning

Firstly, we will show how to obtain the optimal settings of k_p and k_w. The measure setting we take here is $F_{NMED}(w)$, as shown in Formula (12). Again, we choose only one seed word "坑爹(reverse one's expectation)", and the number of words returned is set to $K = 300$. Results in Table 5 show that the performance drops consistently across different k_w settings when the number of patterns increases. Note that at the early stage of Algorithm 1, larger k_p (perhaps with noisy patterns) may lead to lower quality of new words; while larger k_w (perhaps with noisy seed words) may lead to lower quality of lexical patterns. Therefore, we choose the optimal setting to small numbers, as $k_p = 5, k_w = 10$.

Secondly, we justify whether the proposed algorithm is sensitive to the number of seed words. We set $k_p = 5$ and $k_w = 10$, and take F_{NMED} as the weighting measure of new word. We experimented with only one seed word, two, three, and four seed words, respectively. The results in Table 6 show very stable performance when different numbers of seed words are chosen. It's interesting that the performance is totally the same with different numbers of seed words. By looking into the pattern set and the selected words at each iteration, we found that the pattern set ($\mathcal{P}$) converges soon to the same set after a few iterations; and at the beginning several iterations, the selected words are almost the same although the order of adding the words is different. Since the algorithm will finally sort the words at step (11) and $\mathcal{P}$ is the same, the ranking of the words becomes all the same.

Lastly, we need to decide the optimal number of patterns in $\mathcal{P}_c$ (that is, k_c in Algorithm 1) because the set has been used in computing left pattern entropy, see Formula (4). Too small size of $\mathcal{P}_c$ may lead to insufficient estimation of left pattern entropy. Results in Table 7 shows that larger $\mathcal{P}_c$ decrease the performance, particularly when the number of words returned (K) becomes larger. Therefore, we set $|\mathcal{P}_c| = 100$.

4.5 Polarity Prediction of New Sentiment Words

In this section, we attempt to classifying the polarity of the annotated 323 new words. Two methods are adapted with different settings for this purpose. The first one is majority vote (MV), and the second one is pointwise mutual information, similar to (Turney and Littman, 2003). The majority vote method is formulated as below:

$$MV(w) = \sum_{w_p \in PW} \frac{\#(w, w_p)}{|PW|} - \sum_{w_n \in NW} \frac{\#(w, w_n)}{|NW|}$$

where PW and NW are a positive and negative set of emoticons (or seed words) respectively, and $\#(w, w_p)$ is the co-occurrence count of the input word w and the item w_p. The polarity is judged according to this rule: if $MV(w) > th_1$, the word w is positive; if $MV(w) < -th_1$ the word negative; otherwise neutral. The threshold th_1 is manually tuned.

And PMI is computed as follows:

$$PMI(w) = \sum_{w_p \in PW} \frac{PMI(w, w_p)}{|PW|} - \sum_{w_n \in NW} \frac{PMI(w, w_n)}{|NW|}$$

where $PMI(x, y) = log_2(\frac{Pr(x,y)}{Pr(x)*Pr(y)})$, and $Pr(\cdot)$ denotes probability. The polarity is judged according to the rule: if $PMI(w) > th_2$, w is positive; if $PMI(w) < -th_2$ negative; otherwise neutral. The threshold th_2 is manually tuned.

As for the resources PW and NW, we have three settings. The first setting (denoted by

k_w \ k_p	2	3	4	5	10	20	50
5	0.753	0.738	0.746	0.741	0.741	0.734	0.715
10	0.753	0.738	0.746	0.741	0.741	0.728	0.712
15	0.753	0.738	0.746	0.741	0.754	0.734	0.718
20	0.763	0.738	0.744	0.749	0.749	0.735	0.717

Table 5: Parameter tuning results for k_p and k_w. The measure setting is $F_{NMED}(w)$, the seed word set is {"坑爹(reverse one's expectation)"}, and the number of words returned is $K = 300$.

$\#\ seeds \Rightarrow$	1	2	3	4
K=100	0.907	0.907	0.907	0.907
K=200	0.808	0.808	0.808	0.808
K=300	0.741	0.741	0.741	0.741
K=400	0.709	0.709	0.709	0.709
K=500	0.685	0.685	0.685	0.685

Table 6: Performance with different numbers of seed words. The measure setting is $F_{NMED}(w)$, and $k_p = 5$, $k_w = 10$. The seed words are chosen from Table 1.

Large_Emo) is a set of most frequent 36 emoticons in which there are 21 positive and 15 negative emoticons respectively. The second one (denoted by Small_Emo) is a set of 10 emoticons, which are chosen from the 36 emoticons, as shown in Table 8. The third one (denoted by Opin_Words) is two sets of seed opinion words, where PW={高兴(happy),大方(generous),漂亮(beautiful), 善良(kind), 聪明(smart)} and NW ={伤心(sad),小气(mean),难看(ugly), 邪恶(wicked), 笨(stupid)}.

The performance of polarity prediction is shown in Table 9. In two-class polarity classification, we remove neutral words and only make prediction with positive/negative classes. The first observation is that the performance of using emoticons is much better than that of using seed opinion words. We conjecture that this may be because new sentiment words are more frequently co-occurring with emoticons than with these opinion words. The second observation is that three-class polarity classification is much more difficult than two-class polarity classification because many extracted new words are nouns such as "基友(gay)","菇凉(girl)", and "盆友(friend)". Such nouns are more difficult to classify sentiment orientation.

4.6 Application of New Sentiment Words to Sentiment Classification

In this section, we justify whether inclusion of new sentiment word would benefit sentiment classification. For this purpose, we randomly sampled and annotated 4,500 Weibo posts that contain at least one opinion word in the union of the Hownet [4] opinion lexicons and our annotated new words. We apply two models for polarity classification. The first model is a lexicon-based model (denoted by $Lexicon$) that counts the number of positive and negative opinion words in a post respectively, and classifies a post to be positive if there are more positive words than negative ones, and to be negative otherwise. The second model is a SVM model in which opinion words are used as feature, and 5-fold cross validation is conducted.

We experiment with different settings of Hownet lexicon resources:

- Hownet opinion words (denoted by Hownet): After removing some obviously inappropriate words, the left lexicons have 627 positive opinion words and 1,038 negative opinion words, respectively.
- Compact Hownet opinion words (denoted by cptHownet): we count the frequency of the above opinion words on the training data and remove words whose document frequency is less than 2. This results in 138 positive words and 125 negative words.

Then, we add into the above resources the labeled new polar words(denoted by NW, including 116 positive and 112 negative words) and the top 100 words produced by the algorithm (denoted by $T100$), respectively. Note that the lexicon-based model requires the sentiment orientation of each dictionary entry [5], we thus manually label the po-

[4] http://www.keenage.com/html/c_index.html.

[5] This is not necessary for the SVM model. All words in the top 100 words can be used as feature.

$\|\mathcal{P}_c\| \Rightarrow$	50	100	200	300	400	500
K=100	0.907	0.905	0.916	0.916	0.888	0.887
K=200	0.808	0.810	0.778	0.776	0.766	0.764
K=300	0.741	0.731	0.722	0.726	0.712	0.713
K=400	0.709	0.708	0.677	0.675	0.656	0.655
K=500	0.685	0.683	0.653	0.646	0.626	0.627

Table 7: Tuning the number of patterns in $\mathcal{P}_c$. The measure setting is $F_{NMED}(w)$, $k_p = 5$, $k_w = 10$, and the seed word set is {"坑爹(reverse one's expectation)"}.

Emoticon	Polarity	Emoticon	Polarity
	positive		negative
	positive		negative
	positive		negative
	positive		negative
	positive		negative

Table 8: The ten emoticons used for polarity prediction.

Methods $\Rightarrow$	Majority vote	PMI
Two-class polarity classification		
Large_Emo	0.861	0.865
Small_Emo	0.846	0.851
Opin_Words	0.697	0.654
Three-class polarity classification		
Large_Emo	0.598	0.632
Small_Emo	0.551	0.635
Opin_Words	0.449	0.486

Table 9: The accuracy of two/three-class polarity classification.

larity of all top 100 words (we did NOT remove incorrect new word). This results in 52 positive and 34 negative words.

Results in Table 10 show that inclusion of new words in both models improves the performance remarkably. In the setting of the original lexicon (Hownet), both models obtain 2-3% gains from the inclusion of new words. Similar improvement is observed in the setting of the compact lexicon. Note, that $T100$ is automatically obtained from Algorithm 1 so that it may contain words that are not new sentiment words, but the resource also improves performance remarkably.

5 Conclusion

In order to extract *new sentiment words* from large-scale user-generated content, this paper proposes a fully unsupervised, purely data-driven, and

	# Pos/Neg	Lexicon	SVM
Hownet	627/1,038	0.737	0.756
Hownet+NW	743/1,150	0.770	0.779
Hownet+T100	679/1,172	0.761	0.774
cptHownet	138/125	0.738	0.758
cptHownet+NW	254/237	0.774	0.782
cptHownet+T100	190/159	0.764	0.775

Table 10: The accuracy of polarity classfication of Weibo post with/without new sentiment words. N-W includes 116/112 positive/negative words, and T100 contains 52/34 positive/negative words.

almost knowledge-free (except POS tags) framework. We design statistical measures to quantify the utility of a lexical pattern and to measure the possibility of a word being a new word, respectively. The method is almost free of linguistic resources (except POS tags), and does not rely on elaborated linguistic rules. We conduct extensive experiments to reveal the influence of different statistical measures in new word finding. Comparative experiments show that our proposed method outperforms baselines remarkably. Experiments also demonstrate that inclusion of new sentiment words benefits sentiment classification definitely.

From linguistic perspectives, our framework is capable to extract *adjective* new words because the lexical patterns usually modify adjective words. As future work, we are considering how to extract other types of new sentiment words, such as *nounal* new words that can express sentiment.

Acknowledgments

This work was partly supported by the following grants from: the National Basic Research Program (973 Program) under grant No. 2012CB316301 and 2013CB329403, the National Science Foundation of China project under grant No. 61332007 and No. 60803075, and the Beijing Higher Education Young Elite Teacher Project.

References

Shlomo Argamon, Ido Dagan, and Yuval Krymolowski. 1998. A memory-based approach to learning shallow natural language patterns. In Proceedings of the 17th International Conference on Computational Linguistics - Volume 1, COLING '98, pages 67--73, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fan Bu, Xiaoyan Zhu, and Ming Li. 2010. Measuring the non-compositionality of multiword expressions. In Proceedings of the 23rd International Conference on Computational Linguistics, COLING '10, pages 116--124, Stroudsburg, PA, USA. Association for Computational Linguistics.

Keh-Jiann Chen and Wei-Yun Ma. 2002. Unknown word extraction for chinese documents. In Proceedings of the 19th International Conference on Computational Linguistics - Volume 1, COLING '02, pages 1--7, Stroudsburg, PA, USA. Association for Computational Linguistics.

Aitao Chen. 2003. Chinese word segmentation using minimal linguistic knowledge. In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing - Volume 17, SIGHAN '03, pages 148--151, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yaacov Choueka. 1988. Looking for needles in a haystack or locating interesting collocation expressions in large textual databases. In Proceeding of the RIAO'88 Conference on User-Oriented Content-Based Text and Image Handling, pages 21--24.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. Comput. Linguist., 16(1): 22--29, March.

J Ferreira da Silva and G Pereira Lopes. 1999. A local maxima method and a fair dispersion normalization for extracting multi-word units from corpora. In Sixth Meeting on Mathematics of Language, pages 369--381.

Gaël Dias, Sylvie Guilloré, and José Gabriel Pereira Lopes. 2000. Mining textual associations in text corpora. 6th ACM SIGKDD Work. Text Mining.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. Comput. Linguist., 19(1):61--74, March.

Zhen Hai, Kuiyu Chang, and Gao Cong. 2012. One seed to find them all: Mining opinion features via association. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM '12, pages 255--264, New York, NY, USA. ACM.

John S Justeson and Slava M Katz. 1995. Technical terminology: some linguistic properties and an algorithm for identification in text. Natural language engineering, 1(1):9--27.

Hongqiao Li, Chang-Ning Huang, Jianfeng Gao, and Xiaozhong Fan. 2005. The use of svm for chinese new word identification. In Natural Language Processing--IJCNLP 2004, pages 723--732. Springer.

Pavel Pecina. 2005. An extensive empirical study of collocation extraction methods. In Proceedings of the ACL Student Research Workshop, ACLstudent '05, pages 13--18, Stroudsburg, PA, USA. Association for Computational Linguistics.

Fuchun Peng, Fangfang Feng, and Andrew McCallum. 2004. Chinese segmentation and new word detection using conditional random fields. In Proceedings of the 20th International Conference on Computational Linguistics, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. Computational linguistics, 37(1):9--27.

Patrick Schone and Daniel Jurafsky. 2001. Is knowledge-free induction of multiword unit dictionary headwords a solved problem. In Proc. of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2001), pages 100--108.

Richard Sproat and Thomas Emerson. 2003. The first international chinese word segmentation bakeoff. In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing - Volume 17, SIGHAN '03, pages 133--143, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xu Sun, Houfeng Wang, and Wenjie Li. 2012. 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 - Volume 1, ACL '12, pages 253--262, Stroudsburg, PA, USA. Association for Computational Linguistics.

Beijing Thesaurus Research Center. 2003. Xinhua Xin Ciyu Cidian. Commercial Press, Beijing.

Peter D. Turney and Michael L. Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. ACM Trans. Inf. Syst., 21(4):315--346, October.

Hua-Ping Zhang, Hong-Kui Yu, De-Yi Xiong, and Qun Liu. 2003. Hhmm-based chinese lexical analyzer ictclas. In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing -

Volume 17, SIGHAN '03, pages 184--187, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wen Zhang, Taketoshi Yoshida, Xijin Tang, and Tu-Bao Ho. 2009. Improving effectiveness of mutual information for substantival multiword expression extraction. Expert Systems with Applications, 36(8):10919--10930.

Yan Zhang, Maosong Sun, and Yang Zhang. 2010. Chinese new word detection from query logs. In Advanced Data Mining and Applications, pages 233--243. Springer.

Yabin Zheng, Zhiyuan Liu, Maosong Sun, Liyun Ru, and Yang Zhang. 2009. Incorporating user behaviors in new word detection. In Proceedings of the 21st International Jont Conference on Artifical Intelligence, IJCAI'09, pages 2101--2106, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

GuoDong Zhou. 2005. A chunking strategy towards unknown word detection in chinese word segmentation. In Natural Language Processing--IJCNLP 2005, pages 530--541. Springer.