Chinese Sentiment Analysis Using Appraiser-Degree-Negation Combinations and PSO

Wen Xiong[1,2]
[1] Institute of Chinese Information Processing, Beijing Normal University,
19#, WaiDa St., XinJieKou, HaiDian Area, Beijing, China
[2] China Patent Information Center,
25#, BeiTaiPingZhuang Road, HaiDian Area, Beijing, China
Email: xiongwen@cnpat.com.cn

Yaohong Jin and Zhiying Liu
Institute of Chinese Information Processing, Beijing Normal University,
19#, WaiDa St., XinJieKou, HaiDian Area, Beijing, China
Email: {jinyaohong, liuzhy}@bnu.edu.cn

***Abstract*—To recognize effectively the polarity of Chinese sentences using Chinese appraisers, degree adverbs, and negations, this article presents a new rule-based method. First, the method combines three types Chinese words into the pre-defined rules; it employs the word distances of those rules as constraints; it utilizes the strengths of appraisers and degree adverbs as items of the rules. Then, it utilizes the particle swarm optimization (PSO) to obtain the optimal parameters of the rules, such as the thresholds of constraints and the adjustments of the strengths. Furthermore, it uses Chinese lexicon HowNet as the resources of Chinese sentiment. Experiments show that the method realizes higher precision, recall, and F1 than the baseline of C-support vector classification (C-SVC) on two real-world applications. Moreover, it obtains the second rank for the task 2 of Chinese opinion analysis and evaluation (COAE) in 2011, indicating the availability of the method.**

***Index Terms*—sentiment analysis, text mining, particle swarm optimization, support vector machine**

I. Introduction

The researches for sentiment analysis are concerned more and more, which are especially with growing requirements in multiple fields for commercial organizations and governmental departments. Public response for emergent events and consumers' feedback need the tools used for text sentiment classification (TSC) and opinion mining to analyze the sentiment polarity of texts, which can help to discover the problems and the requirements. Thus, TSC becomes a sub task in text mining (TM), and a necessary method for sentiment analysis and opinion mining.

The methods of sentiment analysis can be divided into multiple categories, such as sentiment classification using sentiment phrases (SCSP) [1], sentiment classification using the methods of machine learning (SCML) [2], and sentiment classification using scoring functions (SCSF) [3].

There are methods using phrases or semantic triples as basic constituents for the analysis of the sentence sentiment, which are based on dependency parser, and semantic parser. However, these methods are dependent on the corpus, which are tagged manually to form a training data set. The performance of the parser can be influenced when running in the practical texts due to that the coverage ratio from the training data set is limited. On the other hand, those methods using algorithms of machine learning are influenced due to not tackling sentiment words. In this paper, we propose a rule-based method to analyze the sentence sentiment using sentiment words, and optimize those parameters in the rules using particle swarm optimization (PSO).

The rest of the paper is organized as follows: in Section 2, we describe the role of appraisers, degree adverbs, and negations in the sentiment analysis. Then, we propose the scoring rules using the combinations of appraisers, and the degree adverbs (ADA) and appraisers, degree adverbs, and negations (ADAN) in Section 3. And then, we propose a method combining appraisers, degree adverbs, and negations (ADN) in Section 4, followed by the parameter optimization using PSO. In Section 5, we show the experiments on multiple data sets, such as an open data set, a manually constructed data set, and an evaluative data set used for Chinese opinion analysis and evaluation (COAE)[1]. Section 6 introduces the related work, and Section 7 gives the conclusions.

II. Role Analysis of Sentiment Words

A. Appraiser Roles

Some Chinese words have clear appraisive meaning. Therefore, these words express the main sentiment in the sentence. Due to their different types in part of speech

Manuscript received October 18, 2013; revised October 22, 2013.
Corresponding author: Wen Xiong; email to: xiongwen@cnpat.com.cn

[1] http://ir-china.org.cn/coae2011.html

doi:10.4304/jcp.9.6.1410-1417

(POS), we provide their roles in the sentiment analysis according to their POS as follows:

1) Positive appraiser verbs:

Positive appraiser verbs increase the positive sentiment of Chinese sentences. For example, "Ta Zhenxi Yu Taren Jiaoliu De Jihui (he cherished the opportunity to communicate with others.)" Here are a few words, such as Zhenxi (to cherish), Baojiang (to praise and to honor), and Fenli (to make a long arm).

2) Negative appraiser verbs:

Negative appraiser verbs increase the negative sentiment of Chinese sentences. For example, "Ta Mushi Yu Taren Jiaoliu De Jihui (he ignored the opportunity to communicate with others.)" Here are a few words, such as Moshi (to ignore), Aichi (to lament), and Anhai (to murder).

3) Positive appraiser nouns:

Positive appraiser nouns express the positive sentiment of Chinese sentences. For example, "Ta Yinwei Nuli Gongzuo Huode Liao Biaoyang (he works hard to gain recognition.)" Here are a few words, such as Aimu (adoration), Anhao (ease), and Anggui (costliness).

4) Negative appraiser nouns:

Negative appraiser nouns express the negative sentiment of Chinese sentences. For example, "Ta Kan Qilai Xiang Ge Baichi (he looks like an idiot.)" Here are a few words, such as Angzhang (dinginess), Anggui (costliness), and Baichi (idiot).

5) Positive appraiser Adjectives:

Positive appraiser adjectives increase the sentiment of the modified noun. For example, "Ta Shi Yi Ge Hao Nanren (he is a good man.)" Here are a few words such as Anquan (safe), Bang (good), and Biyao (essential).

6) Negative appraiser adjectives:

Negative appraiser adjectives increase the sentiment of the modified nouns. For example, "Ta Shi Yi Ge Huai Nanren (he is a bad man.)" Here are a few words, such as Buan (unsafe), Zhaogao (bad), and Duoyu (redundant).

7) Positive appraiser adverbs:

Positive appraiser adverbs increase the sentiment of the modified verbs. For example, "Ta Cengken Di Shuo (he said honestly.)" Here are a few words, such as Anshi (schedule), Buchuo (correctly), and Cengzhi (cordially).

8) Negative appraiser adverbs:

Negative appraiser adverbs increase the sentiment of the modified verbs. For example, "Ta Baibai di Huafei Liao Shijian (he spent time in vain.)" Here are a few words, such as Anzhong (secret), Aoman (haughtily), and Baibai (in vain).

9) Positive appraiser idioms:

Positive appraiser idioms express the positive sentiment of Chinese sentences. For example, "Renmen Anjuleye (people live and work in peace and contentment.)" Here are a few words, such as Aibushishou (fondle admiringly), and Anjuleye (live and work in peace and contentment).

10) Negative appraiser idioms:

Negative appraiser idioms express the negative sentiment of Chinese sentences. For example, "Tamen Renwei Ta Aishouaijiao (they think he is in a hindrance.)" Here are a few words, such as Eyufengcheng (greasiness), Aishouaijiao (be in a hindrance), and bamianlinglong (be smooth and slick).

B. Degree Adverb Roles

Chinese degree adverbs can reinforce the sentiment of the modified appraisers, though degree adverbs do not have a negative meaning or positive meaning. They can be divided into six levels by empirically. Therefore, we assigned the degree adverbs of level 1 with weight 10, level 2 with weight 8, level 3 with weight 6, level 4 with weight 4, level 5 with weight -3, and level 6 with weight -1. The positive weight means reinforcement. The bigger weight the stronger reinforcement has a degree adverb. On the other hand, the negative weight means weakening the modified words, the bigger absolute weight the weaker modification has a degree adverb.

We describe their roles as follows:

1) Degree adverbs with reinforcement:

a) Words of level 1 (such as Duo (excessively), Guo (over), and Qiang (strongly)) reinforce the sentiment strength with weight 10.

b) Words of level 2 (such as BaiFenzhibai (absolutely), Jidu (to the utmost), and chongfen (plenty)) reinforce the sentiment strength with weight 8.

c) Words of level 3 (such as Cen (deep), Chuqi (unusually), and hao (well)) reinforce the sentiment strength with weight 6.

d) Words of level 4 (Geng (still), Gengjia (much more), and Nayang (in that way)) reinforce the sentiment strength with weight 4.

2) Degree adverbs with weakening:

a) Words of level 5 (Buda (not too), Liao (a little), and Xiangdui (rather)) weaken the sentiment strength with weight -3.

b) Words of level 6 (Haosheng (properly), Huoduohuoshao (more or less), and Man (quite)) weaken the sentiment strength with weight -1.

C. Negation Roles

Negations reverse the sentiment of appraisers, for example, "Ta Bushi Huai Nanren (he is not a bad man.)" Here are a few words, such as Bingfei (no), Meiyou, (without), and Bugou (not enough).

III. Scoring Rules

A. Rules for ADA

In fact, the strength of appraisers is a fuzzy number, and the number of appraisers is big. Therefore, we can assign a weight 0.4 to the positive appraisers, and a weight -0.4 to the negative appraisers by empirically. On the other hand, the combination of words with different roles has a decisive role for the sentiment polarities.

For example, a Chinese *ADA* combination can be a form as 'da_1 da_2 ap… da_n' with modifications ($0 \le n \le 4$), where '*da*' is the degree adverbs, and '*ap*' is the appraisers. For example, BaiFenzhibai Zhenxi 'absolutely cherishes', Fenwai Chuqi Zhenxi 'especially unusually cherishes', and Zhenxi De Budeliao 'cherishes extremely'.

We developed rules to score an *ADA* by combining both appraisers' weight and degree adverbs' weight for different *ADA*s. The thresholds of the word distances (from t_1 to t_3) are the number of words between two words plus one. When the distance exceeds a threshold, e.g., four, the constraints between two words are treated as invalid. Furthermore, we introduced adjustable integer values for the appraisers and the degree adverbs due to the optimization purpose, which reflects the adjustment amount of constraints of the rules.

$$\text{if } (num(da)=0 \text{ and } polarity(ap)=positive) \Rightarrow \\ score = weight(ap). \tag{1}$$

Where the *score* shows the sentiment of Chinese sentences when the number of *da* (*num (da)*) is zero.

$$\text{if } (num(da)=1 \text{ and } polarity(ap)=positive) \Rightarrow \\ score = \begin{cases} weight(ap), \text{ if } dist(da,ap) > t_1 \\ weight(ap)\cdot(1+weight(da)/c_1), \text{ otherwise.} \end{cases} \tag{2}$$

Where *dist(da, ap)* is the word distance between *da* and *ap*; t_1 is the threshold of the word distance. C_1 is the adjustable integer mentioned above for the modification of the positive appraiser. Therefore, this rule shows *da* reinforces the score of the *ap* when only one *da* (*num (da) = 1*) appears, with the distance between them less than t_1.

$$\text{if } (num(da)=2 \text{ and } polarity(ap)=positive) \Rightarrow \\ score = \begin{cases} weight(ap)\cdot(1+weight(da_1)/c_1), \\ \quad \text{if } pos(da_1) < pos(ap) \text{ and } dist(da_1,ap) \le t_1 \\ score + weight(ap)\cdot(1+weight(da_2)/c_1), \\ \quad \text{if } pos(da_2) < pos(ap) \text{ and } dist(da_2,ap) \le t_1. \end{cases} \tag{3}$$

Where *pos(w)* is the position of the word *w*. Therefore, this rule shows *da*s reinforce the score of the *ap* when two *da*s precede *ap* with the distances between them less than t_1.

$$\text{if } (num(da)>2 \text{ and } polarity(ap)=positive) \Rightarrow \\ score = weight(ap)\cdot(1+(weight(da_1)+weight(da_2))/ \\ c_1\cdot num(da)/2). \tag{4}$$

Equation (4) shows *da*s reinforce the score of the *ap* with the number of the *da*s greater than two.

$$\text{if } (num(da)=0 \text{ and } polarity(ap)=negative) \Rightarrow \\ score = -weight(ap) - c_2. \tag{5}$$

Wherein: c_2 is the adjustable integer for the negative appraiser.

$$\text{if } (num(da)=1 \text{ and } polarity(ap)=negative) \Rightarrow \\ score = \begin{cases} -weight(ap) - c_2, \text{ if } dist(da,ap) > t_1 \\ \quad -weight(ap) - (weight(ap)+c_3)\cdot \\ \quad weight(da)/c_1, \text{ otherwise.} \end{cases} \tag{6}$$

Wherein: c_3 is the adjustable integer for the negative appraiser when the *da* modifies the *ap*. Therefore, this rule shows *da* reinforces the negative score of the *ap* when only one *da* appears, with the distance between them less than t_1.

$$\text{if } (num(da)>2 \text{ and } polarity(ap)=negative) \Rightarrow \\ score = -weight(ap) - (weight(ap)+c_3)\cdot \\ (weight(da_1)+weight(da_2))/c_1\cdot num(da)/2. \tag{7}$$

Equation (7) shows that *da*s reinforce the negative score of the *ap* with the number of the *da*s greater than two.

$$\text{if } (num(da)=2 \text{ and } polarity(ap)=negative) \Rightarrow \\ score = \begin{cases} -weight(ap) - (weight(ap)+c_3)\cdot weight(da_1)/c_1, \\ \quad \text{if } pos(da_1) < pos(ap) \text{ and } dist(da_1,ap) \le t_1, \\ score - weight(ap) - (weight(ap)+c_3)\cdot weight(da_2)/c_1, \\ \quad \text{if } pos(da_2) < pos(ap) \text{ and } dist(da_2,ap) \le t_1 \end{cases} \tag{8}$$

Equation (8) shows *da*s reinforce the negative score of the *ap* when two *da*s precede the *ap* with the distances between them less than t_1.

B. Rules for ADAN

An *ADAN* is a sequence of words: "w_1 w_2 w_3 w_4 *ap* w_5 w_6 … w_8," where the w_i ($1 \le i \le 8$) is degree adverbs or negations, e.g., "Wei Baifenzhibai Zhenxi (not absolutely cherishes)", and "Wei Zhenxi De Budeliao (not cherishes extremely)".

We developed rules to score an *ADAN* by combining appraisers' weight, degree adverbs' weight, and negations' reversion. Those rules for *ADAN*s are as follows.

$$\text{if } (num(ne)=1 \text{ and } num(da)=0 \text{ and} \\ polarity(ap)=positive) \Rightarrow \\ score = \begin{cases} weight(ap), \text{ if } dist(ne,ap) > t_2 \\ -weight(ap) - c_2, \text{ otherwise.} \end{cases} \tag{9}$$

Wherein: *ne* is a negation, and t_2 is the threshold of word distance between *ne* and *ap*. Therefore, the rule shows that the negation reverses the sentiment polarity of the original appraiser with the conditional threshold t_2.

$$\begin{aligned} &\text{if } (num(ne)=1 \text{ and } num(da)=1 \text{ and} \\ &\ polarity(ap)=positive) \Rightarrow \\ &\text{if } (dist(ne,da) \le t_3 \text{ and } pos(da) > pos(ne)) \\ &\quad score = \begin{cases} weight(ap), \text{ if } dist(ne,ap) > t_2 \\ -weight(ap) + (1 - weight(ap) - c_2)\cdot \\ weight(da)/c_1, \text{ otherwise} \end{cases} \\ &\text{elseif } (dist(ne,da) \le t_3 \text{ and } pos(da) < pos(ne)) \\ &\quad score = \begin{cases} weight(ap), \text{ if } dist(ne,ap) > t_2 \\ \quad -weight(ap) - (weight(ap)+c_3)\cdot \\ \quad weight(da)/c_1, \text{ otherwise} \end{cases} \\ &\text{else} \\ &\quad score = \begin{cases} weight(ap), \text{ if } dist(ne,ap) > t_2 \\ -weight(ap) - c_2, \text{ otherwise.} \end{cases} \end{aligned} \tag{10}$$

Equation (10) shows the reinforcing role of *da*, which is either after or before *ne* with the thresholds of t_2 and t_3, when only one *ne* and one *da* appears.

$$\begin{aligned}&\text{if } (num(ne)=2 \text{ and } ((pos(ne_1)>pos(ap)\\&\text{and } pos(ne_2)>pos(ap)) \text{ or } (pos(ne_1)<pos(ap)\\&\text{and } pos(ne_2)<pos(ap))) \text{ and } polarity(ap)=positive) \Rightarrow\\&score=\begin{cases}weight(ap), \text{ if } num(da)=0\\ weight(ap)\cdot(1+weight(da)/c_1),\\ \text{if } num(da)=1.\end{cases}\end{aligned} \tag{11}$$

Equation (11) shows that the two *ne*s must be either prior or posterior the appraiser together.

$$\begin{aligned}&\text{if } (num(ne)=1 \text{ and } num(da)=0 \text{ and } polarity(ap)=negative) \Rightarrow\\&score=\begin{cases}-weight(ap)-c_2, \text{ if } dist(ne,ap)>t_2\\ weight(ap), \text{ otherwise.}\end{cases}\end{aligned} \tag{12}$$

Equation (12) is an allelomorph of (9), indicating the reverse role of the *ne* with the threshold t_2.

$$\begin{aligned}&\text{if } (num(ne)=1 \text{ and } num(da)=1 \text{ and } polarity(ap)=negative) \Rightarrow\\&\text{if } (pos(da)>pos(ne) \text{ and } dist(ne,da)\le t_3)\\&score=\begin{cases}-weight(ap)-c_2, \text{ if } dist(ne,ap)>t_2\\ \quad weight(ap)-(weight(ap)-c_4)\cdot\\ \quad weight(da)/c_1, \text{ otherwise}\end{cases}\\&\text{elseif } (pos(da)<pos(ne) \text{ and } dist(ne,da)\le t_3)\\&score=\begin{cases}-weight(ap)-c_2, \text{ if } dist(ne,ap)>t_2\\ weight(ap)\cdot(1+weight(da))/c_1, \text{ otherwise}\end{cases}\\&\text{else}\\&score=\begin{cases}-weight(ap)-c_2, \text{ if } dist(ne,ap)>t_2\\ weight(ap), \text{ otherwise.}\end{cases}\end{aligned} \tag{13}$$

Equation (13) is an allelomorph of (10), indicating the reinforcing role of *da*. *Da* is either after or before *ne*, where only one *ne* and one *da* appear, with a threshold t_2 or t_3, and with an adjustable integer c_4 for *ne*.

$$\begin{aligned}&\text{if } (num(ne)=1 \text{ and } num(da)>1 \text{ and } polarity(ap)=negative) \Rightarrow\\&score=-weight(ap)-c_5.\end{aligned} \tag{14}$$

Equation (14) is a complementary rule for (12) and (13), and c_5 is an adjustable integer for this case.

$$\begin{aligned}&\text{if } (num(ne)=2 \text{ and } ((pos(ne_1)>pos(ap)\\&\text{and } pos(ne_2)>pos(ap)) \text{ or } (pos(ne_1)<pos(ap)\\&\text{and } pos(ne_2)<pos(ap))) \text{ and } polarity(ap)=negative) \Rightarrow\\&score=\begin{cases}-weight(ap)-c_5, \text{ if } num(da)=0\\ -weight(ap)-(weight(ap)+c_6)\cdot\\ weight(da)/c_1,\\ \quad \text{if } num(da)=1.\end{cases}\end{aligned} \tag{15}$$

Equation (15) is an allelomorph of (11). C_6 is an adjustable integer for this case.

IV. METHODS

To compare our method with machine learning based method for sentiment analysis of Chinese sentences, we used a support vector machine (SVM) method as the baseline classifier, e.g., C-support vector classification (C-SVC) with linear kernel [4].

A. SVM for Sentiment Analysis

The data set can be represented as: $\{x_i, y_i\}$, $y_i \in \{-1,1\}$, $x_i \in R^d$, i=1,…,n, where d is the dimension of word vector, and n is the number of sentences. We formulate SVM solution for nonlinear machines on non-separable data set as follows:

$$\begin{aligned}&\max_{\alpha}\Big(\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{n}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)\Big)\\&\text{s.t. } \sum_{i=1}^{n}y_i\alpha_i=0,\\&\quad 0\le\alpha_i\le C,\ i=1,\dots,n.\end{aligned} \tag{16}$$

Where *K()* is a kernel function, and α_i is a Lagrange multiplier. *C* is a constant. The decision function can be expressed as follows:

$$\begin{aligned}&f(\mathbf{x})=\mathrm{sgn}\Big(\sum_{i=1}^{n}y_i\alpha_i K(\mathbf{x}_i,\mathbf{x})+b^*\Big)\\&\text{s.t. } b^*=y_j-\sum_{i=1}^{n}y_i\alpha_i^* K(\mathbf{x}_i,\mathbf{x}_j),\ 0<\alpha_j^*<C.\end{aligned} \tag{17}$$

We used the bag-of-word (BOW) model to form sentence vectors, and employed term frequency and inverse document frequency (TF-IDF) as the feature values of the vector, which can be expressed as follows:

$$TF\text{-}IDF(t,d)=(1+\log(tf(t,d)))\cdot\log(N/df(t)). \tag{18}$$

Where tf (t, d) is a frequency of term t in a document d, and df (t) is a document frequency of term t. N is the total number of documents.

B. ADN-Scoring for Sentiment Analysis

Based on the rules from (1) to (15), we offer a new method called ADN-scoring to calculate the sentiment polarities of Chinese sentences. The processing steps of the method are as follows:

ADN-scoring algorithm for Chinese sentences:

1. Divide paragraphs into sentences according to the Chinese punctuations, e.g., full stop, exclamatory mark, and question mark;

2. Divide those sentences into clauses according to other Chinese punctuations, e.g., comma, semicolon, pause mark;

3. For each clause:

3.1 Segment it using forward maximum segmentation to obtain word vector and position of words;

3.2 For each word in the word vector:

3.2.1 If the word is an *ap*, check other words in the word vector;

3.2.1.1 When the word is a *ne*, record the position of the *ne* if the number of the *ne*s is less than three; otherwise, do not record it;

3.2.1.2 When the word is a *da*, record the position and weight of the *da* if the number of the *da*s is less than four; otherwise, do not record them;

3.2.2 If the word is a positive *ap*:

3.2.2.1 When no *ne* appears in the word vector, calculate the score of the clause according to the rules from (1) to (4);

3.2.2.2 When only a *ne* appears in the word vector, calculate the score of the clause according to the rules from (9) to (10);

3.2.2.3 When two *ne*s appear in the word vector, calculate the score of the clause according to (11);

3.2.3 When the word is a negative *ap*:

3.2.3.1 When no *ne* appears in the word vector, calculate the score of the clause according to the rules from (5) to (8);

3.2.3.2 When only a *ne* appears in the word vector, calculate the score of the clause according to the rules from (12) to (14);

3.2.3.3 When two *ne*s appear in the word vector, calculate the score of the clause according to the rule (15);

4. Accumulate the scores of the clauses to form the score of the sentence;

5. Return the score of the sentence.

C. Parameter Optimization Using PSO

As mentioned above, the rules from (2) to (15) use parameters: $c_1 \ldots c_6$, t_1, t_2, and t_3. If those parameters are determined manually, uncertain factor will be introduced in them. Therefore, we can use PSO algorithm to optimize those parameters.

PSO was proposed by Kennedy and Eberhart in 1995 [5], simulating the behavior mechanism of swarm, e.g. bird swarm. When bird swarm migrates, or searches food, birds adjust themselves adaptively to obtain optimal targets according to their individual and social knowledge.

Each bird is called a particle, which is treated as an agent. It consults the optima found earlier, and that found by its neighbors, searching the optimal targets by adjusting its position and its velocity in flight.

The PSO algorithm integrates information from the current, the historical, and the neighboring of every particle, forming a stochastic, uniform, and adaptive behavior mechanism.

The particle changes its velocity when it moves, which can be expressed as follows:

$$v_{p,d} = w \cdot v_{p,d} + c_1 \cdot r_1 \cdot (pb_{p,d} - x_{p,d}) + c_2 \cdot r_2 \cdot (gb_d - x_{p,d}). \quad (19)$$

Where $v_{p,d}$ is the velocity of particle p in dimension d, and w is the weight for the original velocity. $x_{p,d}$ is the position of the particle p in dimension d. $pb_{p,d}$ is the personal-best position, and the $gb_{p,d}$ is the global-best position found by the swarm. c_1 and c_2 are two constants for the convergence of swarm; r_1 and r_2 are two random values.

After the update of the velocity, particle changes its position too, which can be expressed as follows.

$$x_{p,d} = x_{p,d} + v_{p,d}. \quad (20)$$

Due to that the velocity and position of each particle can be out of the range of the solution's space, we used a function clamping them, which can be expressed as follows.

$$\begin{aligned} &\text{if } (x_{p,d}^{\max} < x_{p,d})\ x_{p,d} = x_{p,d}^{\max},\ v_{p,d} = 0 \\ &\text{if } (x_{p,d} < x_{p,d}^{\min})\ x_{p,d} = x_{p,d}^{\min},\ v_{p,d} = 0. \end{aligned} \quad (21)$$

V. EXPERIMENTS

We used three data sets from the real-world applications to examine the validity of the proposed ADN-scoring algorithm. First data set was collected manually[2] about a movie called 'Slumdog Millionaire'. We extracted the first paragraph from 566 different Chinese reviews. Then, we filtered the neutral Chinese sentences to form the data set (No.1). The characteristic of the data set is summarized in Table 1.

TABLE I.
CHARACTERISTICS OF DATA SET (NO.1)

Positive Sentence	Negative Sentence	Total
493	296	789

To cover more Chinese sentences using appraisers, we extracted more appraisers from Chinese news Website[3] to extend the appraiser lexicon from HowNet[4]. The characteristic of the appraiser lexicon is summarized in Table 2.

TABLE II.
CHARACTERISTICS OF LEXICON AND EXTENSION

lexicon	Positive appraiser	Negative appraiser
HowNet	4566	4370
extended	3288	4407

Furthermore, we extracted 29 representative Chinese negations manually, which are listed in Table 3.

TABLE III.
CHINESE NEGATIONS EXTRACTED

Negations			
Mei	Bingfei	Meiyou	Bu
Bugou	Buqu	Buhui	Buneng
Bukeneng	Bingbu	Congwei	Fei
Hebu	Buyu	Bushi	Haobu
Bushi	Benbu	Haowu	Shiqu
Juebu	Juefei	Juebu	Meishen mo
Wei	Wu	Yaobu	Yongbu
Yuanfei	-	-	-

We used the popular measures to evaluate the ADN-scoring algorithm, such as precision, recall, and F1, which are expressed as follows:

$$\begin{aligned} &precision = tp/(tp+fp) \\ &recall = tp/(tp+fn) \\ &F_1 = \frac{2 \cdot recall \cdot precision}{recall + precision} = \frac{2 \cdot tp}{2 \cdot tp + fn + fp}. \end{aligned} \quad (22)$$

[2] http://movie.mtime.com/80325/comment.html
[3] http://news.sina.com.cn
[4] http://www.keenage.com

Where:

1) *tp* is true positive, equaled to the number of positive cases judged to positive category;

2) *fp* is false positive, equaled to the number of negative cases judged to positive category;

3) *fn* is false negative, equaled to the number of positive cases judged to negative category.

The sentiment is positive when the score is greater than zero, and is negative when the score is less than zero.

We ran PSO algorithm five times to determine the parameters of the PSO as follows:

1) Each of particles has nine dimensions, corresponding to the nine parameters of the ADN-scoring algorithm, i.e., $c_1 \ldots c_6, t_1, t_2, t_3$.

2) Each run has 20 iterations.

3) The w is 0.6893, and the c_1 and c_2 of the PSO algorithm are 1.

4) The r_1 and r_2 are uniform random numbers in the range $[0, w]$.

5) The fitness of particle is F1.

Cross-validation (CV) is a popular evaluative method in the training and the test procedures. We used data set No.1 as the training data set, and 3-fold CV to calculate the F1 values.

In our method, the size of the neighbor is the size of the swarm. The position of every particle is initialized to a uniform random value in the solution's space. Similarly, the velocity of every particle is also initialized to another uniform random value within the range between 0 and half of the maximum value of the position.

After running ADN-scoring algorithm, we attained the optimal parameters as follows:

(a) t_1 is 2; (b) t_2 is 5; (c) t_3 is 3; (d) c_1 is 17; (e) c_2 is 0.177074; (f) c_3 is 0.074365; (g) c_4 is 1; (h) c_5 is 0.622489; (i) c_6 is 0.657678.

The results of SVM baseline and ADN-scoring using the 3-fold CV on the data set No.1 are listed in Table 4.

TABLE IV.
RESULTS OF SVM AND ADN ON DATA SET NO.1

Method	Precision	Recall	F1
SVM	0.8746	*0.8579*	0.8662
ADN	*0.9227*	0.8414	*0.8799*

From Table 4, ADN-scoring algorithm realized slightly better than SVM baseline.

We extracted three typical examples, in which SVM classifier failed, but ADN method succeeded as follows.

1) Yiyang de pin (they are similarly poor.) The length of the Chinese sentence is short, where 'pin' is a negative appraiser. SVM algorithm failed in this short sentence due to the word is not existed in the training data set, but ADN method recognized successfully the appraiser 'pin', which is included in the lexicon.

2) Zhuduo jiangxiang, sihu youxie yanguoqishi (many awards, it seems some exaggeration.) In this Chinese sentence, 'yanguoqishi' is a negative appraiser (a Chinese idiom). Many words in this sentence do not appear in the training data. Therefore, SVM algorithm failed in this case, but ADN method successfully recognized the idiom due to having a large lexicon, including that idiom.

3) Wo juede meiyou bu gei manfen de liyou (there is no reason not to give full points.) In this sentence, 'meiyou' and 'bu' are two negations, and 'manfen' is a positive appraiser. The double negations only can be recognized by rules. Although SVM algorithm failed in this case, ADN succeeded due to that the lexicon includes those negations.

Furthermore, we adopted more complicated data sets called corpus-sentiment[5]. The characteristics of the data sets are summarized in Table 5.

TABLE V.
CHARACTERISTICS OF DATA SETS (NO.2)

Data set	Positive articles	Negative articles	Total articles
Hotel	2000	2000	4000
Notebook	2000	2000	4000
Book	2000	2000	4000

Each article in the data sets can have both positive and negative sentences, which increases the difficulty of sentiment classification. We calculated the sum of sentiment polarities of those sentences in the article as the final polarity. Thus, we ran the SVM baseline and ADN-scoring algorithm on this data set (No. 2) using 3-fold CV with the parameters optimized on the training data set (No. 1). The results are as follows.

TABLE VI.
RESULTS OF SVM AND ADN-SCORING METHOD ON DATA SETS NO.2

Method	Data set	Precision	Recall	F_1
ADN	Hotel	0.7550	*0.815*	*0.7838*
	Notebook	*0.7333*	*0.8675*	*0.7948*
	Book	*0.7332*	*0.808*	*0.7688*
SVM	Hotel	*0.8212*	0.5071	0.6267
	Notebook	0.7203	0.5462	0.6209
	Book	0.6840	0.5280	0.5953

From table 6, the results of ADN-scoring are better than that of the SVM baseline.

We guessed that the noise in the data reduced the performance of SVM due to that an effective attribute selection or feature extraction was not adopted before using SVM baseline, which is normally an important pre-processing step for SVM method.

Furthermore, we used the closest data set of evaluation task two in COAE to evaluate, and calculated the macro-

[5] www.searchforum.org.cn/tansongbo/corpus-senti.htm

and micro-evaluation results for the three fields (digital, entertainment, and finance). The results are as follows.

TABLE VII.
RESULTS FOR CLASSIFICATION OF OPINION SENTENCES

Method	Macro Evaluation			Micro Evaluation		
	Prec	Recall	F1	Prec	Recall	F1
HITSZ	*0.5347*	*0.7234*	*0.5414*	*0.4945*	0.6544	*0.6254*
ADN	0.3315	0.7195	0.4086	0.3994	*0.7618*	0.5241
Suda	0.3706	0.4054	0.3840	0.3652	0.5048	0.5089

From table 7, we see that the ADN approach attained better results than the Suda method which obtained the second rank.

VI. Related Work

Sentiment and subjectivity classification can be run on two levels: document, and sentence [6]. The former mostly uses supervised learning methods. Naive Bayesian and SVM are such algorithms used for the sentiment classification. Moreover, feature selection can improve the performance of these algorithms [7]. The latter are normally treated as a middle procedure of the former. Supervised learning methods are also can be used for the latter [8]. Meantime, the methods of semi-supervised learning [9] and the methods of opinion word collection [10] have also been effective. Especially, the context has been influential too [11], such as negations and contraries. In [12], the strength of opinion and types of subjectivity are classified to neutral, low, medium, or high using deeply nested clauses within a sentence. Another target of sentiment classification is discovering the objects of opinion first, and then deciding their sentiment polarities [13].

The model of sentiment composition [14] can classify the sentiment of the grammatical constituents, which used a quasi-compositional way, and was implemented as a post-process of lexical parsing, interpreting the output of a dependency parser. In this paper, instead of acquiring the linguistic constituents on the grammatical levels, we used constraints existed in between the words closed to each other, and expressed those constraints by using word distances, where the appraisers are the center of sentence sentiments.

An analytical approach to assess text sentiment was proposed in [15]. The method processed the semantics, followed by employing rules to assign contextual valences to the linguistic components. Instead of acquiring the valences of Chinese linguistic components, we calculated the sentence sentiment directly using three types of words, and utilizing constraints, combinations, and rules with their adjustable parameters.

The method learning with compositional semantics was presented in [16], which incorporated structural inference, and motivated by integrating the compositional semantics into learning procedure. Experiments showed the performance was better (90.7%) than that using simple heuristics based on compositional semantics (89.7%), and that using learning-based methods without compositional semantics (89.1%). In the method, the constraints of word distance were ignored.

A method of sentiment classification for Japanese and English sentences was proposed in [17], which was based on conditional random fields (CRF) with hidden variables and dependency tree to compute the polarity of the entire sentence, where the cooperation of the hidden variables was considered.

AVA method [18] analyzed the sentiment strength of English sentences using adjectives, verbs, and adverbs with their combinations, where two AVAC-scoring algorithms have been developed.

In the paper, the method is different from the AVA. First, it processed sentences in Chinese instead of that in English. Second, it used appraises, degree adverbs, negations, and their combinations, where those appraisers included multiple POSs, such as verbs, nouns, adjectives, adverbs, and idioms, and those degree adverbs were only part of the whole adverbs. Third, the method employed word distance constraints, Chinese lexicons from HowNet, weighted adjustments, and parameterized rules.

A method called Elastic Net was proposed for online sentiment classification [19], which used character-level, word-level n-grams, and different combinations of them to improve the interpretability. Experiments showed that it outperformed semantic-oriented approaches according to the accuracy of sentiment classification. The top-ranked features were selected, which had strong sentiment polarities, and had significant implications for markets.

In [20], a framework was introduced for Chinese documents, which combined context-sensitive sentiment lexicon and sentiment words from Hownet to improve the performance of opinion mining.

In [21], a new method of opinion mining was proposed, which used hierarchical fuzzy domain sentiment ontology to define a space of product features and opinions. Therefore the user experience was improved, and the experimental results showed that the polarity of many sentiment expressions was automatically identified by the approach.

VII. Conclusions

We present a ADN-scoring method in the paper, which used appraisers, degree adverbs, negations, and their combinations for the sentiment analysis of Chinese sentence. Furthermore, We utilized PSO algorithm to optimize the parameters of the rules for the method. When those parameters are determined by the PSO algorithm on the training corpus, the performance of the sentiment classification for new sentences and texts will be improved. The experimental results indicate the improvement on new test data sets. The model can be extended to new applications. For example, it can be applied to the social media, such as Blog and Micro-Blogging.

The sentiment words used in this article came from general fields. Therefore, those words can be used in other fields. However, there are a few sentiment words for specific fields not including in our lexicons. On the other hand, some words' sentiment will change in different contexts and situations. For example, the Chinese word Laji 'junk' can be ambiguous. When it used for a certain person, it is a negative term meaning as 'bad person'. Otherwise, it is a neutral term meaning as 'void items.' Those ambiguous words are useful for special fields. Furthermore, the lexicons in the method can be expanded. How to integrate the fields into the lexicons to identify a certain sentiment in those fields more precisely will be the future work.

ACKNOWLEDGMENT

This work is supported by "863: the Hi-Tech Research and Development Program of China (No. 2012AA011104)."

REFERENCES

[1] P. Turney, "Thumbs up or Thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews," Proc. The Meeting of the Association for Computational Linguistics (ACL'02), pp. 417-424, 2002.

[2] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification Using Machine Learning Techniques," Proc. The EMNLP. pp. 79-86, 2002.

[3] K. Dave, S. Lawrence, and D. Pennock, "Mining the Peanut Gal-lery: Opinion Extraction and Semantic Classification of Product Reviews," Proc. The 12th Intl. World Wide Web Conference (WWW'03), pp. 519-528, 2003.

[4] C.C. Chang and C.J. Lin, "LIBSVM: A Library for Support Vector Machines," Available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

[5] J. Kennedy, and R. Eberhart," Particle Swarm Optimization," Proc. ICNN. vol. 4, pp. 1942-1948, 1999.

[6] Bing Liu. "Sentiment Anlaysis and Subjectivity." Invited Chapter for the *Handbook of Natural Language Processing*, Second Edition (editors: N. Indurkhya and F. J. Damerau), March, 2010.

[7] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis." *Foundations and Trends in Information Retrieval*, 2(1-2), pp. 1–135, 2008.

[8] H. Yu and V. Hatzivassiloglou, "Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences," Proc. The Conference on Empirical Methods in Natural Language Processing (EMNLP), 2003.

[9] M. Gamon, A. Aue, S. Corston-Oliver, and E. Ringger, "Pulse: Mining Customer Opinions from Free Text," Proc. The International Symposium on Intelligent Data Analysis (IDA), pp. 121–132, 2005.

[10] S.-M. Kim and E. Hovy, "Determining the Sentiment of Opinions," Proc. The International Conference on Computational Linguistics (COLING), 2004.

[11] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing Contextual Polarity in Phrase-level Sentiment Analysis," Proc. The Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pp. 347–354, 2005.

[12] T. Wilson, J. Wiebe, and R. Hwa, "Just How Mad are You? Finding Strong and Weak Opinion Clauses," Proc. AAAI. pp. 761–769, 2004.

[13] B. Liu, M. Hu, and J. Cheng, "Opinion Observer: Analyzing and Comparing Opinions on the Web," Proc. WWW. 2005.

[14] K. Moilanen and S. Pulman, "Sentiment Composition," Proc. The Recent Advances in Natural Language Processing International Conference (RANLP 2007), pp. 378–382, 2007.

[15] M. Al Masum Shaikh, H. Prendinger, and M. Ishizuka, "An Analytical Approach to Assess Sentiment of Text," Proc. ICCIT. 2007.

[16] Y. Choi and C. Cardie, "Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis," Proc. EMNLP. pp. 793-801, 2008.

[17] T. Nakagawa, K. Inui, and S. Kurohashi, "Dependency Tree-Based Sentiment Classification using CRFs with Hidden Variables," Proc. ACL: HLT, 2010.

[18] V.S. Subrahmanian and D. Reforgiato, "AVA: Adjective-Verb-Adverb Combinations for Sentiment Analysis," *IEEE Intelligent System*, vol. 23, no. 4, pp. 43-50, 2008.

[19] L. Fan, Y.L. Zhang, and H.C. Chen, "Sentiment Analysis on Chinese Web Forums using Elastic Nets: Features, Classification and Interpretation," FCB Working Paper Series, 2011.

[20] L. Liu, M. Lei, and H. Wang, "Combining Domain-Specific Sentiment Lexicon with Hownet for Chinese Sentiment Analysis." *Journal of Computers*, vol. 8, no. 4, pp. 878-883, 2013.

[21] H. Wang, X. Nie, L. Liu, and J. Lu, "A Fuzzy Domain Sentiment Ontology based Opinion Mining Approach for Chinese Online Product Reviews," *Journal of Computers*, vol. 8, no. 9, pp. 2225-2231, 2013.

Wen Xiong was born in Wuhan City, Hubei Province, in 1968. He received the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 2011. From January 2011 to April 2013, he was a postdoctoral fellow with the Institute of Chinese Information Processing, Beijing Normal University, Beijing, China. Since May 2013, he has been an employee at China Patent Information Center, and CPIC-BNU joint laboratory of machine translation. His research interests include machine translation, natural language processing, artificial intelligence, affective computing, and data mining.

Yaohong Jin received the MA and Ph.D. degrees from Chinese Academy of Sciences, P. R. China. He is a professor at the Institute of Chinese Information Processing, Beijing Normal University, China. Previously, he was an associate professor of Chinese Academy of Sciences. His research interests include natural language processing, machine translation and search engine, in which areas he has published more than 20 papers in international journals and conference proceedings.