Identifying Evaluative Sentences in Online Discussions

Zhongwu Zhai[†] Bing Liu[‡] Lei Zhang[‡] Hua Xu[†] Peifa Jia[†]

[†]State Key Lab of Intelligent Tech. & Sys., Tsinghua National Lab for Info. Sci. and Tech.,
Dept. of Comp. Sci. & Tech., Tsinghua Univ.
{zhaizw06@mails, xuhua@mail, dcsjpf@mail}.thu.edu.cn

[‡]Dept. of Comp. Sci., University of Illinois at Chicago
{liub, lzhang3}@cs.uic.edu

Abstract

Much of opinion mining research focuses on product reviews because reviews are opinion-rich and contain little irrelevant information. However, this cannot be said about online discussions and comments. In such postings, the discussions can get highly emotional and heated with many emotional statements, and even personal attacks. As a result, many of the postings and sentences do not express positive or negative opinions about the topic being discussed. To find people's opinions on a topic and its different aspects, which we call *evaluative opinions*, those irrelevant sentences should be removed. The goal of this research is to identify evaluative opinion sentences. A novel unsupervised approach is proposed to solve the problem, and our experimental results show that it performs well.

1. Introduction

Opinion mining aims to find people's opinions/sentiments about topics and aspects/features of the topics (Hu and Liu 2004; Liu 2010). Much of the current research has been focused on extracting opinions from *product reviews* (Pang and Lee 2008; Liu 2010). A key characteristic of reviews is that each review is dedicated to the evaluation of a specific product. There is little interaction among reviewers or irrelevant content. However, this is not the case for online discussions or comments. In such discussions, besides opinions on topics there are typically many other types of postings as the participants can interact with each other. In many cases, the discussions can get emotionally charged and off topic. For example, in our data sets, 66% of the sentences are emotional, abusive or other types. For opinion mining that needs people's opinions on topics and their aspects, which we call *evaluative opinions*, such non-evaluative sentences need to be identified. For example, in a soccer match, the comment "*The German defense is strong*" is a piece of evaluative opinion because it praises the defense (one aspect) of the German team. However, "*I feel so sad for Argentina.*" and "*you know nothing about defense!*" are not evaluative opinions because they do not comment on any aspect of the game or the players.

Opinion mining of online discussions is important, perhaps even more important than mining reviews, because such discussions often focus on current events and issues (they normally have no reviews), and the latest products (their reviews often come much later). Our work was also motivated by some applications in a startup company, where the users only want evaluative opinions. Note that we do not claim that emotional statements are not useful. In fact, they can be useful in some other applications, e.g., finding fans and the mood of the fans. For example, the author of the sentence "*I feel so sad for Argentina*" is likely to be an Argentina *fan*, and his/her mood is *sad*.

The goal of this work is to identify evaluative (opinion) sentences. To our knowledge, this problem has not been studied before. Although it may look similar to subjectivity classification, as we will see later it is entirely different. This paper does not further classify the sentiment in each evaluative sentence as there are existing works for the purpose (Yu and Hatzivassiloglou 2003; Wilson, Wiebe and Hwa 2004; Wiebe and Riloff 2005; Kim and Hovy 2006a).

Clearly, our problem is a classification problem with 2 classes, *evaluative* and *non-evaluative*. The classic approach is supervised learning. However, this approach is hard to scale due to the time-consuming manual labeling of training data. Different applications also need different training data to be labeled. In this paper, we propose a novel unsupervised approach, which only needs a set of *evaluative opinion words* and a set of *emotion words*, which are available. Evaluative opinion words, which we also call *evaluation words*, are words that are often used in evaluations, e.g., *beautiful*, *expensive*, and *ugly*. Emotion words are words that are used to express people's emotions, e.g., *sad*, *surprise*, and *anger* (Parrott 2001). Note that *opinion words* used in the current literature in fact contain both evaluation words and some emotion words, e.g., *sad* and *anger*, but not *surprise* (as *surprise* does not indicate an opinion). In this work, we treat them separately. Note that due to these input word lists (called *lexicons*), one can say that our method is not fully unsupervised, but *weakly semi-supervised*. For simplicity and because of the availability of these word lists, we call the proposed method unsupervised. It is based on 2 important observations.

1. An evaluative opinion should comment on a topic or some aspects of a topic. For example, the evaluative opinion "*The German team was strong*" comments on the aspect "*German team*". Thus topics and aspects are good indicators of evaluative opinions and should be discovered. For easy presentation, we will use the term *aspects* to mean both topics and aspects from now on.
2. Evaluation words and emotion words are indications of evaluative and emotional sentences, respectively. For

example, "*sad*", which is an emotion word in the above example, indicates that the sentence is an emotional sentence. Thus, we need a list of evaluation words and emotion words. However, none of the available such words lists are complete. Hence, they need to be expanded based on the domain corpus.

We uses a similar method as that in (Qiu *et al.* 2010) to extract aspects and to expand the given evaluation and emotion word lists automatically. We then propose a classification technique that only uses the extracted aspects, evaluation words and emotion words. This method (called *A-E-Lexi* in Section 4) actually works reasonably well.

However, we can do much better by exploiting inter-relationships of these concepts to deal with some shortcomings of the *A-E-Lexi* algorithm.

1. A sentence containing an aspect can be an emotional sentence, an evaluative sentence or any other type of sentence. For example, "*I felt sad for the German team*", which contains an aspect "*German team*", is not an evaluative sentence, but "*German team was weak today*" is an evaluative sentence. It turns out in each domain some aspects can be associated with both evaluative opinions and emotions, while others are almost exclusively associated with evaluative opinions. We thus need a method to compute a score for each aspect according how strongly it is associated with evaluative opinions. Note that non-evaluative sentences not only contain emotional sentences, but also many other types of sentences. However, the other types are easier to deal with because they usually do not contain evaluation words and/or aspects.
2. The original lists of evaluation words and emotion words can have errors because the same words may take on different meanings in different domains. We need a method to fix the errors based on a domain corpus.

A novel method is proposed to solve these problems by exploiting that the inter-relationships of the three concepts (aspects, evaluation words, and emotion words). That is, the co-occurrence of an aspect and an evaluation word reinforce each other. The co-occurrence of an emotion word and an aspect inhibit each other. These relationships can be defined circularly and solved iteratively to assign a score to each term representing how strongly it indicates an evaluative opinion. The resulting scores are used to perform the final classification. Our experimental study was based on four Chinese datasets, which are discussion postings of four different topics. The results demonstrated the effectiveness of the proposed method.

2. Related Work

Our work is most related to sentence level opinion mining, more specifically, subjectivity classification (Yu and Hatzivassiloglou 2003; Wilson, Wiebe and Hwa 2004; Wiebe and Riloff 2005; Pang and Lee 2008), which determines whether a sentence is subjective or objective. However, evaluative sentences are different from subjective sentences because many subjective sentences are not evaluative sentences. For example, the sentence "*I feel so sad for Argentina.*" is a subjective sentence, but is not an evaluative sentence. It is actually an emotional sentence.

Our work is also related to (Kim and Hovy 2006a), which analyzes judgment opinions. Opinions are of two main kinds: (1) beliefs about the world, with values such as true, false, possible, unlikely, etc.; (2) judgments about the world, with values such as good, bad, neutral, wise, foolish, virtuous, etc. The statement "*I believe that Germany played badly today*" is an example of a belief whereas "*Germany played very well today*" is a judgment opinion. In our definition, we treat both these two examples as evaluative opinion sentences as their classification can be quite subjective. Furthermore, no technique to identify judgment opinions was proposed in (Kim and Hovy 2006a). They also did not find any topic or aspect. In (Kim and Hovy 2006b), the authors proposed a supervised method to find reasons for pros and cons, which is different from our work as we do not find such reasons, and also our technique is unsupervised. In (Hassan, Qazvinian and Radev), a method is proposed to identify the attitudes of participants toward one another in online discussions. That is, it predicts whether a sentence displays an attitude toward a text recipient. Our work is again different as we focus on evaluative sentences.

3. The Proposed Technique

Figure 1 gives an overview of the proposed technique. Given the raw discussion postings, the algorithm works in 4 steps to identify *evaluative* sentences in the postings:

Pre-processing: Each posting is segmented into sentences by period, question and exclamation marks. Each sentence is also POS-tagged. Since this step is fairly simple, it will not be discussed further in this paper.

Extraction of aspects and expansion of evaluation and emotion lexicons: This step is discussed in Section 3.1.

Interaction modeling of aspects, evaluation words and emotion words: This step is described in Section 3.2.

Classification: This step is discussed in Section 3.3.

3.1 Extraction of Aspects and Expansion of Evaluation and Emotion Lexicons

This section presents the technique for discovering aspects and expanding the given evaluation word list and the given emotion word list. In this work, we use a method similar to the *double propagation* (*DP*) method in (Qiu *et al.* 2010). *DP* is a bootstrapping technique. It uses some dependency relationships of opinion words and aspects to extract as-

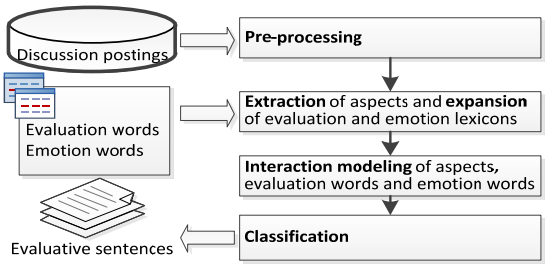

Figure 1. Overview of the proposed technique

pects and expand the initial seed opinion words iteratively. The input is only a list of opinion words. However, since we are interested in evaluation words and emotion words separately, we need to modify the *DP* method.

The main idea of the *DP* method can be illustrated by the following sentence:

"*The phone has good screen*."

In the dependency tree, we can find that "good" modifies "screen". Then, if "good" is known to be an opinion word, "screen" can be extracted as an aspect. If "screen" is known to be an aspect, then "good" will be extracted as an opinion word. Here, the "modifying" relationship is used for mutual extraction.

A key step of the *DP* method is to build an accurate dependency tree. Since we are interested in Chinese text, we need a Chinese dependency parser. To our knowledge, there are three main dependency parsers for Chinese, i.e., ICTParser[1], LTP[2] and Stanford Parser[3]. ICTParser and LTP are Web demos and not ready for others to use. We experimented with the Stanford Parser. However, it did not perform well. We thus could not use it. Instead, we make use of POS tags to approximate the relations in (Qiu *et al.* 2010) for our purpose, which we discuss below.

Our adapted technique performs the following tasks:

1. Extract *aspects* using *evaluation* or *emotion words*;
2. Extract *aspects* using extracted *aspects*;
3. Extract *evaluation* words and *emotion words* using the given or extracted *evaluation words* and *emotion words* respectively.

For each subtask above, different rules are proposed:

Rule for Task 1 (E→A): If a noun term *N* appears near a given or extracted evaluation or emotion word *E*, *N* is extracted as an aspect *A* if there is no adjective or noun terms between *N* and *E*. If two or more noun terms appear near *E*, the nearest noun term is selected. Here the term represents a word or phrase.

This rule is quite intuitive, and is mainly useful for evaluation words as an evaluation is typically expressed on a target aspect. For example, in sentence (a) below, "差(weak)" is the given evaluation word and "阿根廷(Argentina)" and "后防(defense)" are both noun terms, "后防(defense)" is finally detected as the aspect, since "后防(defense)" is nearer to the given evaluation word "差(weak)" than "阿根廷(Argentina)".

a. 阿根廷/n 的/u 后防/n 很/d 差/adj (Argentina defense is very weak.)

Rules for Task 2 (A→A): There are two rules here:

(1) If one of the conjoined noun terms is an extracted aspect, then the other noun term is also an aspect.

In sentence (b), "勒夫(Löw)" and "小伙子们(the players)" are conjoined by the conjunction word "和(and)". Then, if "小伙子们(the players)" has been extracted as an aspect, "勒夫(Löw)" will be extracted as an aspect as well, and vice versa.

[1] http://nlp.ict.ac.cn/demo/ictparser/
[2] http://ir.hit.edu.cn/demo/ltp/
[3] http://nlp.stanford.edu/software/lex-parser.shtml

Input: Text corpus: *R*
Evaluation word seeds: *vas* // the given evaluation word lexicon
Emotion word seeds: *mos* // the given emotion word lexicon
Output: All evaluation words: *VA*
All emotion words: *MO*
All aspects: *A*

```
1: VA = vas;  MO = mos;  A = ∅
2: seedVA = vas;  seedMO = mos;  seedA = ∅
3: while (seedVA != ∅ | seedMO != ∅ | seedA != ∅):
4:     deltaVA = ∅;  deltaMO = ∅;  deltaA = ∅
5:     for each POS-tagged sentence in R:
6:         // Task 1
7:         Extract aspects newA using E→A based on seedVA∪seedMO
8:         Add the elements in newA but not in A into deltaA
9:         // Task 2
10:        Extract aspects newA using A→A based on seedA
11:        Add the elements in newA but not in A into deltaA
12:        // Task 3
13:        Extract evaluation words newVA using E→E based on seedVA
14:        Add the elements in newVA but not in VA∪MO into deltaVA
15:        Extract emotion words newMO using E→E based on seedMO
16:        Add the elements in newMO but not in VA∪MO into deltaMO
17:    Add deltaVA into VA
18:    Add deltaMO into MO
19:    Add deltaA into A
20:    seedVA = deltaVA;  seedMO = deltaMO;  seedA = deltaA
```

Figure 2. Algorithm for discovering aspects and expanding evaluation and emotion words lists

b. 勒夫/n 和/c 小伙子们/n 都/d 很/d 努力/adj (Löw and the players are both hard-working.)

(2) If a noun term *N* appears before or after an extracted aspect *A* and they are separated by "的", then *N* is extracted as an aspect.

Applying this rule to sentence (a), if "后防 (defense)" is an extracted aspect, then "阿根廷 (Argentina)" is inferred as an aspect *A*.

Rules for task 3 (E→E): Again, we have two rules:

(1) If an adjective term *Adj* appears within a text window of three words before or three words after a given or extracted *evaluation* word *E*, then *Adj* is extracted as a new evaluation word.

Take sentence (c) as an example, if "强(strong)" is a given evaluation word, then the adjective term "主动(proactive)" is extracted as a new evaluation word. "主动(proactive)" is classified as an evaluation word since "强(strong)" is an evaluation word.

c. 德国队/n 后防/n 很/d 主动/adj 很/d 强/adj (The German defense is proactive and strong.)

(2) If an adjective term *Adj* appears within a text window of three words before or three words after a given or extracted *emotion* word *E*, *Adj* is extracted as a new emotion word.

We use separate rules because we want to know whether an evaluation or emotion word is extracted.

Based on the above description, the detailed algorithm is given in Figure 2, which is self-explanatory.

3.2 Aspects, Evaluation Words and Emotion Words Interaction

In the above step, we extracted evaluation words, emotion words, and aspects. However, these pieces of information are still insufficient because aspects can appear in both evaluative and non-evaluative sentences, and the original categorization of evaluation words and emotion words may

not be suitable for each particular application domain, which also results in the expanded lists having errors. This section deals with these problems.

Based on the observations in the introduction section, we postulate that aspects and evaluation words are key indicators of evaluative sentences. To deal with possible errors, we want to weight those possibly wrong evaluation words or emotion words down. Since aspects can appear in both emotion sentences and evaluative sentences, we want to weight the aspects that are associated with both evaluation words and emotion words down in order to lower down their effects on the final classification. To formulate the idea, we use the following intuitions:

1. An extracted aspect that is modified by or associated with many *evaluation words* is more likely to indicate an evaluative sentence. Then, we want to give a high score to the aspect.
2. An extracted aspect that is modified by or associated with many *emotion words* is not a good indicator of an evaluative sentence. It should be assigned a low score.
3. A given or extracted evaluation word that does not modify *good* (high scored) aspects are likely to be a wrong evaluation word, and should be weighted down.
4. The more evaluative the aspects are, the less emotional their associated emotion words should be.

We model the relations with a directed tripartite graph in Figure 3. These interactions indicate a circular definition of the three concepts, aspects, evaluation words and emotion words. The definition bears some resemblance to the HITS algorithm in (Kleinberg 1999). The main difference is that we also have emotion words, which behave as *inhibiters*. They do not exist in HITS. This gives us the third layer, emotion words layer. We called the proposed formulation *IAEE* (*Interaction of Aspect, Evaluation and Emotion*).

Formally, the tripartite graph is represented as $G = <V_a, V_{va}, V_{mo}, E_{va\text{-}a}, E_{mo\text{-}a}>$, where $V_a = \{a_i\}$, $V_{va} = \{va_j\}$, $V_{mo} = \{mo_k\}$ are *aspects*, *evaluation words* and *emotion words*, respectively; $E_{va\text{-}a}$ denotes the relationship between V_{va} and V_a; $E_{va\text{-}a}$ denotes the relationships between V_{mo} and V_a.

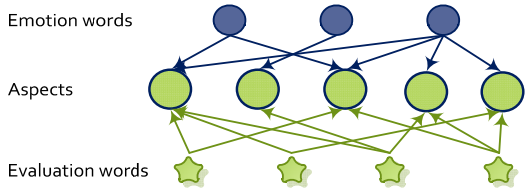

Figure 3. Interaction modeling of aspects, evaluation words and emotion words (IAEE)

Input: Evaluation words: *VA*
Emotion words: *MO*
Aspects: *A*
Co-occurrence relationship between *VA* and *A*: $E_{va\text{-}a}$
Co-occurrence relationship between *MO* and *A*: $E_{mo\text{-}a}$
Output: Evaluative scores of *VA*, *MO* and *A*
1: Initialize the scores of *asp*, *eva* and *emo* to 1;
2: **Repeat** 50 times**:**
3: update each $asp(a_i)$ using Eq. 1
4: update each $eva(va_j)$ using Eq. 2
5: update each $emo(mo_k)$ using Eq. 5
6: normalize $asp(a_1)$, $asp(a_2)$,…, $asp(a_I)$ to [0,1]
7: normalize $eva(va_1)$, $eva(va_2)$,…, $eva(va_J)$ to [0,1]
8: normalize $emo(mo_1)$, $emo(mo_2)$,…, $emo(mo_K)$ to [0,1]

Figure 4. Iterative computation of the IAEE model

Here, the relationship refers to co-occurrence in a sentence. That is, if an aspect $a \in V_a$ and an evaluation (or emotion) word $v \in V_{va}$ (or $m \in V_{mo}$) co-occurs in a sentence, a directed edge (v, a) (or (m, a)) is created. The edges are all of unit weight, i.e., multiple occurrences are considered as 1.

Let *eva* be the *evaluative* score for an evaluation word, and *emo* be the *emotion* score for an emotion word. The score for an aspect *asp* is defined by Eq. 1, where *asp* is positively dependent on the sum of the evaluation word scores, and negatively dependent on the sum of the emotion word scores. Thus, when *asp* is positive, the aspect is evaluative; when *asp* is negative, the aspect is emotional. A parameter λ (*damping factor*) is used to adjust the relative influences of evaluation words and emotion words. In this study, λ is set to the default value 0.5.

$$asp(a_i) = \lambda \times \sum_{(i,j)\in E_{va-a}} eva(va_j) - (1-\lambda) \times \sum_{(i,k)\in E_{mo-a}} emo(mo_k) \quad (1)$$

Since aspects and evaluation words mutually reinforce each other, the score *eva* of an evaluation word is computed with Eq. 2, where a_i is an associated aspect with the evaluation word va_j.

$$eva(va_j) = \sum_{(i,j)\in E_{va-a}} asp(a_i) \quad (2)$$

The computation of score $emo(mo_k)$ for each emotion word mo_k is involved. To consider the inhibiting effect, we first introduce an intermediate score $tmp(mo_k)$, which is defined by Eq. 3. Since the *emo* score indicates non-evaluative strength, the score for an emotion word $emo(mo_k)$ should have opposite effect of $tmp(mo_k)$. That is, the larger the $tmp(mo_k)$ is, the smaller the $emo(mo_k)$ should be as shown in Eq. 4. To achieve the desired effect of $emo(mo_k)$, we define it with Eq. 5, where *max* represents the maximum value of $tmp(mo_k)$ of all emotion words (see Eq. 6).

$$tmp(mo_k) = \sum_{(i,k)\in E_{mo-a}} asp(a_i) \quad (3)$$

$$emo(mo_k) \propto -tmp(mo_k) \quad (4)$$

$$emo(mo_k) = -tmp(mo_k) + max = max - tmp(mo_k) \quad (5)$$

$$max = \max\{tmp(mo_1), tmp(mo_2), \ldots, tmp(mo_{|V_{mo}|})\} \quad (6)$$

To further understand Eqs. 3~6, let us discuss two extreme cases. If $tmp(mo_k)$ is very high, which means that the aspects are strong because it is computed from aspects in Eq. 3, then the emotion score should be low. This is reflected by Eq. 5. The strong aspects are caused by strong connections with evaluation words due to their positive mutual reinforcements. If $tmp(mo_k)$ is very low, which means the aspect scores are low because it is computed in Eq. 2, then the emotion score should be high. This is also reflected by Eq. 5. When emotion words are strong, the aspect associated with them will be pushed down (low aspect score) and vice versa. Eq. 1 does just that.

To solve the equations, we use the classic power iteration method. The detailed algorithm is given in Figure 4. The input includes the text corpus, evaluation words *VA*, emotion words *MO* and aspects *A*. The algorithm outputs individual scores for each aspect, evaluation word and emotion word. We run on each dataset for 50 iterations in our experiments, which is sufficient.

3.3 Classification

Given the scored evaluation words, emotion words, and aspects, and the corpus, this step classifies each sentence.

Task 1: It matches all aspects $\{a_1, \ldots, a_I\}$ in the sentence s and finds the highest evaluative score *topA* of an aspect in s (Eq. 7). If *topA* is greater than a pre-defined threshold ρ (the default is 0.6), we proceed to step 2; Otherwise, the sentence s is classified as non-evaluative.

$$topA = \max_{1 \le i \le I} asp(a_i) \quad (7)$$

For example, for the sentence "*German defense is proactive and strong*", this step first finds the only aspect "*German defense*". Assume its *asp* score's higher than 0.6, we go to step 2.

Task 2: It matches all evaluation words $\{va_1, \ldots, va_J\}$ and emotion words $\{mo_1, \ldots, mo_K\}$ in the sentence, and then sums up the evaluation word scores *vaSum* (Eq. 8) and emotion word scores *moSum* (Eq. 9). If *vaSum* is greater than *moSum*, sentence s is classified as evaluative; Otherwise, non-evaluative.

$$vaSum = \sum_{1 \le j \le J} eva(va_j) \quad (8)$$

$$moSum = \sum_{1 \le k \le K} emo(mo_k) \quad (9)$$

Following the above example, we have two evaluation words "proactive" and "strong", which result in *vaSum* > 0; there is no emotion word, resulting in *moSum* = 0. This sentence is thus classified as an evaluative sentence.

4. Empirical Evaluation

We used 4 datasets to evaluate the proposed *IAEE* system. The datasets were crawled from a popular Chinese news discussion site (http://news.sina.com.cn). The datasets are discussions about (1) 2010 FIFA, (2) 2010 NBA, (3) Guo Degang's dispute with Beijing TV, and (4) Tang Jun's fake PhD degree. Two CS PhD students were employed to annotate all the sentences as evaluative or non-evaluative. The Kappa scores for the inter-rater agreements range from 0.841 to 0.894, which indicate almost perfect agreement. The details of the datasets are given in Table 1.

4.1 Methods and Settings

The proposed algorithm ***IAEE*** is compared with 6 baseline methods, which are categorized into supervised and unsupervised methods. We list the supervised methods first.

NB: It uses a Naive Bayesian classifier.

SVM: It uses the Support Vector Machines.

For NB and SVM, we use Chinese segmented words as features in training and testing. All results are obtained from 10-fold cross validation.

The unsupervised category has the following:

Table 1. Summary of the four datasets

	#Postings	#Sentences		Kappa
		evaluative	non-evaluative	
ARG VS. GER(FIFA)	1672	1393	1607	0.894
Lakers VS. Celtics(NBA)	1984	883	2117	0.881
Guo Degang(GD)	2196	682	2318	0.847
Tang Jun (TJ)	1712	1115	1885	0.841

Lexi: It uses the original evaluation and emotion lexicons from (HowNet). If the number of evaluation words is more than the number of emotion words in a sentence s, s is classified as evaluative; otherwise non-evaluative. If s has no evaluation word, it is non-evaluative.

E-Lexi: It is similar to *E-Lexi*, but the expanded evaluation and emotion words (Section 3.1) are also considered.

A-E-Lexi: It is *E-Lexi* but also employs the extracted aspects (Section 3.1). For a sentence s, if it contains at least one aspect, and has more evaluation words than emotion words, then s is classified as evaluative; otherwise non-evaluative. If s has no aspect or no evaluation word, it is classified as non-evaluative.

Double-HITS: This method is based on the extracted aspects, the given and expanded evaluation words and emotion words (Section 3.1). Two HITS algorithms are run separately: *evaHITS* works on aspects (authorities) and evaluation words (hubs), and *emoHITS* works on aspects (authorities) and emotion words (hubs). We then obtain two scores *evaAuth* and *emoAuth* for each aspect, *evaHub* for each evaluation word, and *emoHub* for each emotion word. In classification, for each sentence s, if s contains aspects $\{a_1, \ldots, a_I\}$, evaluation words $\{va_1, \ldots, va_J\}$ and emotion words $\{mo_1, \ldots, mo_K\}$, and it meets the conditions (10) and (11) below, then s is classified as evaluative; otherwise, non-evaluative.

$$\max_{1 \le i \le I}\{evaAuth(a_i)\} > \max_{1 \le i \le I}\{emoAuth(a_i)\} \quad (10)$$

$$\sum_{1 \le j \le J}\{evaHub(va_j)\} > \sum_{1 \le k \le K}\{evaHub(mo_k)\} \quad (11)$$

4.2 Evaluation Results

The comparison results are shown in Table 2, where Avg represents the average result of the 4 datasets. Below we discuss some detailed observations:

- The F-score of the proposed *IAEE* method is the best overall. It is considerably better than all other methods.
- The fully supervised methods *NB* and *SVM* performed poorly in F-score. We believe the main reason is that the key deciding factors for evaluative sentences are the aspects and evaluation words, but these higher level concepts are hard to detect by the 2 supervised techniques.
- On F-score, *Lexi* is not as good as *E-Lexi*, which is not as good as *A-E-Lexi*. The reason is that *Lexi* does not use any expanded evaluation words and emotion words, but only the original words from HowNet for classification. *E-Lexi* works better than *Lexi* because it also uses the expanded evaluation and emotion words. This shows that the discovery step in Section 3.1 is useful. *A-E-Lexi* is even better as it uses the aspect information as well.
- *Double-HITS* performs better than all methods above it on F-score. We believe that the reason is that it is able to re-weight aspects, evaluation and emotion words which partially deals with the interaction of the three concepts.
- The *IAEE* method and *A-E-Lexi* are similar, but *IAEE* uses the weighted aspects, evaluation words and emotion words. We can see that *IAEE* is much better than *A-E-Lexi*, which shows that the step discussed in Section 3.2 is highly effective. *IAEE* improves the F-score of *A-*

Table 2. Comparison Results

	F-score					Precision					Recall				
	FIFA	NBA	GD	TJ	Avg	FIFA	NBA	GD	TJ	Avg	FIFA	NBA	GD	TJ	Avg
NB	0.76	0.63	0.43	0.61	**0.61**	0.75	0.60	0.48	0.61	**0.61**	0.76	0.66	0.40	0.62	**0.61**
SVM	0.74	0.53	0.30	0.57	**0.53**	0.81	0.77	0.68	0.74	**0.75**	0.68	0.40	0.20	0.46	**0.44**
Lexi	0.68	0.53	0.43	0.59	**0.56**	0.61	0.39	0.29	0.46	**0.44**	0.77	0.84	0.79	0.85	**0.81**
E-Lexi	0.73	0.66	0.52	0.69	**0.65**	0.71	0.52	0.38	0.56	**0.54**	**0.74**	**0.91**	**0.85**	**0.91**	**0.85**
A-E-Lexi	0.74	0.70	0.53	0.76	**0.68**	0.77	0.58	0.39	0.66	**0.60**	0.71	0.89	0.84	0.89	**0.83**
Double-HITS	0.75	0.72	0.59	0.76	**0.70**	0.81	0.64	0.48	0.68	**0.65**	0.70	0.83	0.76	0.86	**0.79**
IAEE	**0.75**	**0.81**	**0.69**	**0.78**	**0.76**	**0.83**	**0.81**	**0.64**	**0.70**	**0.75**	0.67	0.81	0.74	0.88	**0.78**

E-Lexi by 8% on average. We can see that the precision of *A-E-Lexi* is much worse than that of *IAEE*, but recall is better than *IAEE*.

- *IAEE* is also better than *Double-HITS* on F-score. Their final classification strategies are the same. The reason that *IAEE* does better is that it is able to fully consider the interaction of the three concepts in a single framework, while *Double-HITS* consider them separately and thus is unable to take advantage of evaluation and emotion interaction through aspects.

In summary, we can conclude that the proposed *IAEE* method is superior to all the baseline methods.

4.3 Influence of the parameters

The proposed *IAEE* has two parameters: the damping factor λ and the evaluative score threshold ρ. We now show the influences of their values on the overall performance. In Figure 5, when λ is around 0.5 *IAEE* achieves the best results (averages over the 4 datasets), which means that evaluative and emotion words should have the same weight. We used a range of ρ values. They showed similar trends for λ. Figure 5 used ρ = 0.6. In Figure 6, when ρ is 0.6 (with λ = 0.5), IAEE achieves the best F-score.

We believe that these parameters give users the flexibility to tune to suit their needs (e.g., balancing the precision and recall). Although it is desirable to have no parameters, for a complex environment it is very difficult for a fixed algorithm to be the best for all possible applications.

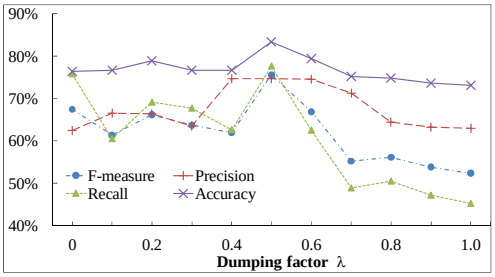

Figure 5. Influence of λ on IAEE

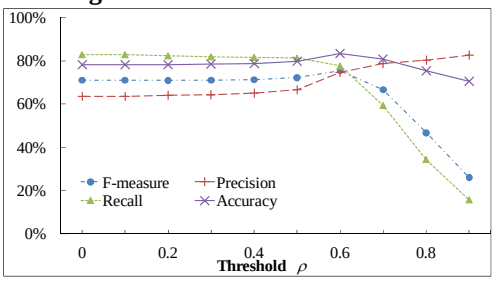

Figure 6. Influence of ρ on IAEE

5. Conclusions

This paper proposes the problem of identifying evaluative sentences from online discussions. To our knowledge, this problem has not been studied. Yet, it is very important for practical applications. We proposed a novel unsupervised method to solve it, which saves the time consuming manual labeling of training data for each application in supervised learning. Extensive experiments based on real-life discussions showed that the proposed method was effective and performed even better than the supervised baselines.

6. Acknowledgments

This work was done when the first author was visiting the University of Illinois at Chicago. He was also partially supported by a grant (Grant No: 60875073) from National Natural Science Foundation of China.

7. References

Hassan A., V. Qazvinian and D. Radev. 2010. *What's with the Attitude? Identifying Sentences with Attitude in Online Discussions*. Proc. of EMNLP.

HowNet.http://www.keenage.com/html/e_bulletin_2007.htm.

Hu M. and B. Liu. 2004. *Mining and Summarizing Customer Reviews*. Proc. of KDD.

Kim S. and E. Hovy. 2006a. *Identifying and Analyzing Judgment Opinions*. Proc. of HLT-NAACL.

Kim S. and E. Hovy. 2006b. *Automatic Identification of Pro and Con Reasons in Online Reviews*. Proc. of COLING.

Kleinberg J. 1999. *Authoritative Sources in a Hyperlinked Environment*. Journal of the ACM 46(5). 46**:** 604-632.

Liu B. 2010. *Sentiment Analysis and Subjectivity*. *Handbook of Natural Language Processing* N. Indurkhya and F. J. Damerau.

Pang B. and L. Lee. 2008. *Opinion Mining and Sentiment Analysis*. *Foundations and Trends in Information Retrieval*. 2**:** 1-135.

Parrott W. 2001. *Emotions in Social Psychology*, Psyc. Press.

Qiu G., B. Liu, J. Bu and C. Chen. 2010. *Opinion Word Expansion and Target Extraction through Double Propagation*. Computational Linguistics.

Wiebe J. M. and E. Riloff. 2005. *Creating Subjective and Objective Sentence Classifiers from Unannotated Texts*. Proc. of CICLing.

Wilson T., J. Wiebe and R. Hwa. 2004. *Just How Mad Are You? Finding Strong and Weak Opinion Clauses*. Proc. of AAAI.

Yu H. and V. Hatzivassiloglou. 2003. *Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences*. Proc. of EMNLP.