Extraction of Opposite Sentiments in Classified Free Format Text Reviews

Dong (Haoyuan) Li[1], Anne Laurent[2], Mathieu Roche[2], and Pascal Poncelet[1]

[1] LGI2P - École des Mines d'Alès, Parc Scientifique G. Besse, 30035 Nîmes, France
E-mail: {Haoyuan.Li, Pascal.Poncelet}@ema.fr
[2] LIRMM - Université Montpellier II, 161 rue Ada, 34392 Montpellier, France
E-mail: {laurent, mroche}@lirmm.fr

Abstract. Opinion mining received well attention in finding personal opinions from user generated content. These opinions contain valuable information for improving and/or comparing the products or services. On the other hand, given a customer review that the opinion has been already classified into a certain sentiment polarity, for instance positive or negative, the opposite sentiments are more and more interesting for the decision makers. In this paper, we propose an unexpected sequence mining based approach to extract opposite sentiments from classified free format text reviews. We first adapt the notions in sequence mining to the opinion mining, then we represent the sentence-level sentiments as the sequential implication rules and from which we generate the belief system for formalizing opposite sentiments. We further propose the algorithm MOSUS (Mining Opposite Sentiments as Unexpected Sentences) that extracts opposite sentiments with respect to the belief system. We conclude by detailing our experimental results on free format text movie review data.

1 Introduction

The opinion mining, that finds personal opinions expressed in user generated content like reviews, forums, discussion groups, blogs, et cetera, received a great deal of attention. Most of existing opinion mining approaches concentrate on the classifications of the sentiment polarities of such user generated content, for instance, the classification of *positive* or *negative* opinions [2, 4, 6, 11]. Such opinions always contain valuable information, for example, in customer opinions on some products, the positive options may help the providers to keep their advantages and the negative options can be important to improve the products. Nevertheless, all those opinions can be useful to new customers for comparing different features between the products.

For mining sentiment polarities from customer reviews, different levels of classifications are considered. At document level, the classification assumes that each review focuses on a single opinion; at the sentence level, the classification is performed to each sentence contained in reviews; at the feature level, the classification extracts sentiments on related features that have been commented

in reviews. In this paper, we focus on mining sentence level sentiment that is opposite in polarity to the classification of the opinion of reviews. For instance, about a notebook computer, a positive review may contain the sentences like "however the graphics performance is not enough", or in a negative review we may also find "anyway this notebook is beautiful". With the growth of classified opinions, this kind of *opposite sentiments* become more and more interesting for the decision makers, since they stand for the information contradicting existing knowledge.

We present our approach for extracting the opposite sentiments from classified reviews in free format text. We first adapt the basic notions in sequence mining [1] to the opinion mining, then we represent the sentence-level sentiments as a set of sequential implication rules. We further create a belief system from these sequential implication rules, and from which we therefore formalize the opposite sentiments. We propose the algorithm MOSUS (Mining Opposite Sentiments as Unexpected Sentences) that extracts opposite sentiments with respect to the belief system.

The rest of this paper is structured as follows. Section 2 is the problem statement, where we formalize the opposite sentiment mining problem. Section 3 presents our approach for extracting opposite sentiments. Section 4 details our experiments on finding opposite sentiments in positive and negative movie-review data. Section 5 is a short conclusion.

2 Problem Statement

We are given a set of free format text reviews that have been already classified into positive or negative opinion polarities. Each review consists in an ordered list of sentences, and each sentence consists in an ordered list of words. We associate the *Part-of-Speech Tag* (PoS tag) [7, 8] with each word, where all PoS tags of the same word category are treated as the same simplified tag. For example, Fig. 1 shows a list of simplified PoS tags of adjectives, adverbs, nouns and verbs.

Category	Simplified PoS Tag	Standard PoS Tags
Adjective	J	JJ, JJR, JJS
Adverb	R	RB, RBR, RBS
Noun	N	NN, NNS, NP, NPS
Verb	V	VB, VBD, VBG, VBN, VBP, VBZ

Fig. 1. Simplified PoS tags of adjectives, adverbs, nouns and verbs.

A *vocabulary*, denoted as v, is a word associated with a simplified PoS tag. For example (be:V) is a vocabulary where be is a word and V is the base tag standing for the verbs. Without loss of generality, we use the wild-card "*" and a simplified PoS tag for denoting a generalized vocabulary. For example, (*:V) denotes a vocabulary that is a verb. Especially, we use (NOT) for denoting the

adverb (`not:R`), or (`n't:R`), so that by default when we say the term *vocabulary*, we do not include (`NOT`). Let $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ be a set of a limited number of distinct vocabularies, a *clause*, denoted as s, is an ordered list of vocabularies $\langle v_1 v_2 \ldots v_k \rangle$. The *length* of a clause is the number of vocabularies contained in the clause, denoted as $|s|$. For example, $\langle (\texttt{film:N})(\texttt{be:V})(\texttt{good:J}) \rangle$ is a clause with length 3, in the order (`film:N`) followed by (`be:V`) and followed by (`good:J`). A vocabulary could also be a clause with length 1 if it is reduced to one word and its associated tag. An *empty clause* is denoted as $\emptyset$, we have $s = \emptyset \iff |s| = 0$. The *concatenation* of clauses is denoted as the form $s_1 \cdot s_2$ so that we have $|s_1 \cdot s_2| = |s_1| + |s_2|$.

Within the context of mining sequence patterns [1], a vocabulary is an *item* and a clause is a *sequence*. Given two clauses $s = \langle v_1 v_2 \ldots v_m \rangle$ and $s' = \langle v'_1 v'_2 \ldots v'_n \rangle$, if there exist integers $1 \leq i_1 < i_2 < \ldots < i_m \leq n$ such that $v_i = v'_{j_i}$ for all v_i, then s is a *sub-clause* of s', denoted as $s \sqsubseteq s'$. If we have $s \sqsubseteq s'$, we say that *s is contained in s'*, or *s' supports s*. If clause s is not contained in any other clauses, then we say that the clause s is *maximal*. For example, the clause $\langle (\texttt{film:N})(\texttt{good:J}) \rangle$ is contained in the clause $\langle (\texttt{film:N})(\texttt{be:V})(\texttt{good:J}) \rangle$, but is not contained in the clause $\langle (\texttt{be:V})(\texttt{good:J})(\texttt{film:N}) \rangle$.

A *sentence*, denoted as S, is a maximal clause that is terminated by one of the following symbols ": , ; . ? !" in the given text. For example, we consider "A nice film, should watch." has two sentences terminated by "," and "." such as $\langle (\texttt{a:DT})(\texttt{nice:J})(\texttt{film:N}) \rangle$ and $\langle (\texttt{should:V})(\texttt{watch:V}) \rangle$. A *document*, denoted as $\mathcal{D}$, is an ordered list of sentences. Given a document $\mathcal{D}$, the *support* or *frequency* of a clause s, denoted as $\sigma(s, \mathcal{D})$, is the total number of sentences $S \in \mathcal{D}$ that support s. Given a user specified threshold of support called *minimum support*, denoted as min_supp, a clause is *frequent* if $\sigma(s, \mathcal{D}) \geq min\_supp$.

We represent the sentiment polarities as the sequential implication rules, and we therefore propose a belief system for formalizing the opposite sentiments expressed in classified documents.

An *sequential implication rule* on clauses, denoted as r, is a rule $s_\alpha \Rightarrow s_\beta$, where s_α and s_β are two clauses; given a clause s, if we have $s_\alpha \cdot s_\beta \sqsubseteq s$, then we say that the clause s *supports* the rule r, denoted as $s \models r$. A *belief* on clauses, denoted as b, consists in an implication rule $s_\alpha \Rightarrow s_\beta$ and a semantical constraint $s_\beta \not\sim s_\gamma$, where the clause s_γ is semantically contradicts the clause s_β. We note a belief as $b = [s_\alpha; s_\beta; s_\gamma]$. A belief constrains that if the clause s_α occurs in a clause s, i.e., $s_\alpha \sqsubseteq s$, then the clause s_β should occur in s after the occurrence of s_β, and the clause s_γ should not occur in s after the occurrence of s_α, that is,

$$[s_\alpha; s_\beta; s_\gamma] \iff s_\alpha \sqsubseteq s \implies s_\alpha \cdot s_\beta \sqsubseteq s \wedge s_\alpha \cdot s_\gamma \not\sqsubseteq s.$$

A clause s that verifies a belief b is *expected*, denoted as $s \models b$; that violates a belief b is *unexpected*, denoted as $s \not\models b$. Given a belief $b = [s_\alpha; s_\beta; s_\gamma]$ and a

clause s such that $s_\alpha \sqsubseteq s$, the unexpectedness is considered as:

$$s_\alpha \cdot s_\beta \not\sqsubseteq s \wedge s_\alpha \cdot s_\gamma \sqsubseteq s \Longrightarrow s \not\models b.$$

Example 1. Given a belief $b = [\langle(\texttt{be:V})\rangle ; \langle(\texttt{good:J})\rangle ; \langle(\texttt{bad:J})\rangle]$ and two clauses $s_1 = \langle(\texttt{be:V})(\texttt{a:DT})(\texttt{good:J})(\texttt{film:N})\rangle$, $s_2 = \langle(\texttt{be:V})(\texttt{bad:J})(\texttt{actor:N})\rangle$, we have $s_1 \models b$ and $s_2 \not\models b$. □

Let M^+ be the positive sentiment and M^- be the negative sentiment, a sentiment $M \in \{M^+, M^-\}$ can be expressed in documents (denoted as $\mathcal{D} \models M$), sentences (denoted as $S \models M$), clauses (denoted as $s \models M$) or vocabularies (denoted as $v \models M$). In addition, we denote the negation of a sentiment M as $\overline{M}$, so that we have $\overline{M^+} = M^-$ and $\overline{M^-} = M^+$. The negation is taken into account in other text-mining applications (for instance for synonym/antonym extraction process [10]).

Proposition 1. *Given a sentiment $M \in \{M^+, M^-\}$, if a document $\mathcal{D} \models M$, then there exists at least one sentence $S \in \mathcal{D}$ such that $S \models M$; if a sentence $S \models M$, then there exists at least one vocabulary $v \sqsubseteq S$ such that $v \models M$ or at least one clause $\langle(\texttt{NOT})v\rangle \sqsubseteq S$ (or $\langle v(\texttt{NOT})\rangle \sqsubseteq S$) such that $v \models \overline{M}$.*

We focus on the sentiments expressed by the sentences that contain adjectives and nouns/verbs, such as "`this is a good job`". Actually, the opinions polarities are often given by the adjectives [10, 3]. The sentiment expressed by sentences like "`this job is well done`" is currently not considered in our approach. Note that we extract basic words relations without the use of syntactic analysis tools [9] to avoid the silence in the data (*i.e.* syntactic relations not extracted by the natural language systems). So that given a sentence $S \models M$, according to Proposition 1, there exists at least one clause $s \sqsubseteq S$ that supports one of the sequential implication rules listed in Fig. 2. We call such a sequential implication rule a *sentiment pattern*. Example 2 shows the sentences that support the sentiment patterns.

Rule Type	Sentiment Rule	Adjective Polarity
JN	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:N})\rangle$	$(\texttt{*:J}) \models M$
NJ	$\langle(\texttt{*:N})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$(\texttt{*:J}) \models M$
VJ	$\langle(\texttt{*:V})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$(\texttt{*:J}) \models M$
JV	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:V})\rangle$	$(\texttt{*:J}) \models M$
NOT-JN	$\langle(\texttt{NOT})(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:N})\rangle$	$(\texttt{*:J}) \models \overline{M}$
NOT-NJ	$\langle(\texttt{*:N})(\texttt{NOT})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$(\texttt{*:J}) \models \overline{M}$
NOT-VJ	$\langle(\texttt{*:V})(\texttt{NOT})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$(\texttt{*:J}) \models \overline{M}$
NOT-JV	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:V})(\texttt{NOT})\rangle$	$(\texttt{*:J}) \models \overline{M}$

Fig. 2. Sentiment patterns for representing the sentiment $M \in \{M^+, M^-\}$.

Example 2. Considering a movie review that has been classified as positive, the following phrases may appear and support the positive polarity of the opinion expressed in this review:

- "`is a good film`" supports the JN pattern $\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{film:N})\rangle$ and the VJ pattern $\langle(\texttt{be:V})\rangle \Rightarrow \langle(\texttt{good:J})\rangle$;
- "`film is good`" supports the NJ pattern $\langle(\texttt{film:N})\rangle \Rightarrow \langle(\texttt{good:J})\rangle$;
- "`a great film moved me much`" supports the JV pattern $\langle(\texttt{great:J})\rangle \Rightarrow \langle(\texttt{move:V})\rangle$;
- "`is not a bad film`" supports the NOT-JN pattern $\langle(\texttt{not:R})(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{film:N})\rangle$ and the NOT-VJ pattern $\langle(\texttt{be:V})(\texttt{not:R})\rangle \Rightarrow \langle(\texttt{bad:J})\rangle$;
- "`film is not bad`" supports the NOT-NJ pattern $\langle(\texttt{film:N})(\texttt{not:R})\rangle \Rightarrow \langle(\texttt{bad:J})\rangle$;
- "`it's a bad film but I don't think so`" supports the NOT-JV pattern $\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{do:V})(\texttt{n't:R})\rangle$.

$\square$

We therefore propose a set of belief patterns for generating the belief base from the sentiment patterns listed in Fig. 2. Figure 3 shows the sentiment patterns and the belief patterns, where the vocabulary (`*-:J`) stands for each antonym of the vocabulary (`*:J`). Each sentence violating a belief generated by one of the patterns stands for a sentence that expresses a potential opposite sentiment.

Rule Type	Sentiment Pattern	Belief Pattern
JN	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:N})\rangle$	$[\langle(\texttt{*-:J})\rangle;\emptyset;\langle(\texttt{*:N})\rangle]$
		$[\langle(\texttt{NOT})(\texttt{*:J})\rangle;\emptyset;\langle(\texttt{*:N})\rangle]$
NJ	$\langle(\texttt{*:N})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$[\langle(\texttt{*:N})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{*-:J})\rangle]$
		$[\langle(\texttt{*:N})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{NOT})(\texttt{*:J})\rangle]$
VJ	$\langle(\texttt{*:V})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$[\langle(\texttt{*:V})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{*-:J})\rangle]$
		$[\langle(\texttt{*:V})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{NOT})(\texttt{*:J})\rangle]$
		$[\langle(\texttt{*:V})(\texttt{NOT})\rangle;\langle(\texttt{*-:J})\rangle;\langle(\texttt{*:J})\rangle]$
JV	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:V})\rangle$	$[\langle(\texttt{*:J})\rangle;\langle(\texttt{*:V})\rangle;\langle(\texttt{*:V})(\texttt{NOT})\rangle]$
NOT-JN	$\langle(\texttt{NOT})(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:N})\rangle$	$[\langle(\texttt{NOT})(\texttt{*-:J})\rangle;\emptyset;\langle(\texttt{*:N})\rangle]$
NOT-NJ	$\langle(\texttt{*:N})(\texttt{NOT})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$[\langle(\texttt{*:N})(\texttt{NOT})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{*-:J})\rangle]$
NOT-VJ	$\langle(\texttt{*:V})(\texttt{NOT})\rangle \Rightarrow \langle(\texttt{*:J})\rangle$	$[\langle(\texttt{*:V})(\texttt{NOT})\rangle;\langle(\texttt{*:J})\rangle;\langle(\texttt{*-:J})\rangle]$
NOT-JV	$\langle(\texttt{*:J})\rangle \Rightarrow \langle(\texttt{*:V})(\texttt{NOT})\rangle$	$[\langle(\texttt{*-:J})\rangle;\emptyset;\langle(\texttt{*:V})(\texttt{NOT})\rangle]$

Fig. 3. Sentiment patterns and belief patterns.

3   Extracting Opposite Sentiments

In this section, we propose our approach MOSUS that extracts the opposite sentiments from classified free format text reviews, with respect to a belief base generated by the sentiment and belief patterns listed in Fig. 3.

Our approach consists of two procedures. We first construct a belief base from a classified learning document, then we find all unexpected sentences from the target documents standing for the opposite sentiments.

3.1   Construction of the Belief Base

Given a set of vocabularies, it is called a *dictionary*. Let $\mathcal{V}$ be a dictionary that contains a set of adjectives expressing the sentiment M. Let $\overline{\mathcal{V}}$ be the dictionary that contains the antonym(s) of each vocabulary contained in $\mathcal{V}$. So that for each adjective $(\texttt{*:J}) \in \mathcal{V}$, we have $(\texttt{*:J}) \models M$ and $(\texttt{*-:J}) \in \overline{\mathcal{V}}$. Now given a *learning document* $\mathcal{D}_L$ such that for each sentence $S \in \mathcal{D}_L$, there exist at least one adjective $(\texttt{*:J}) \in \mathcal{V}$ or there exist $(\texttt{NOT})$ and at least one adjective $(\texttt{*:J}) \in \overline{\mathcal{V}}$.

Let $\mathcal{P}$ denotes the set of sentiment patterns listed in Fig. 3, we have $\mathcal{P} = \{\text{JN, NJ, VJ, JV, NOT-JN, NOT-NJ, NOT-VJ, NOT-JV}\}$, where all those patterns are listed by the descendent priority order. In order to construct the belief base from sentiment patterns, we first apply a sequential pattern mining algorithm for discovering all maximal frequent clauses from $\mathcal{D}_L$ with respect to a minimum support threshold, denoted as $\mathcal{D}_F$. For each clause $s \in \mathcal{D}_F$, we verify whether s supports a sentiment pattern $p \in \mathcal{P}$ with the priority order.

Given $p \in \mathcal{P}$, if $s \models p$, then we add the matched parts $p_M \sqsubseteq s$ to the set $\mathcal{P}_M$, that stands for *instantiated sentiment patterns* that express sentiment M. For describing the notion of instantiated sentiment patterns, let us consider the following example.

Example 3. Given a clause $s = \langle (\texttt{this:DT})(\texttt{be:V})(\texttt{a:DT})(\texttt{good:J})(\texttt{film:N}) \rangle$, we have that s supports the sentiment patterns JN and VJ, and the instantiated sentiment patterns are $\langle (\texttt{good:J}) \Rightarrow (\texttt{film:N}) \rangle$ and $\langle (\texttt{be:V}) \Rightarrow (\texttt{good:J}) \rangle$. We have the priority of JN is higher than VJ, so that $\langle (\texttt{good:J}) \Rightarrow (\texttt{film:N}) \rangle$ is added to the set $\mathcal{P}_M$, even though $\langle (\texttt{be:V}) \Rightarrow (\texttt{good:J}) \rangle$ occurs before the occurrence of $\langle (\texttt{good:J}) \Rightarrow (\texttt{film:N}) \rangle$. □

We can therefore construct a belief base from each pattern $p \in \mathcal{P}_M$ with respect to the belief patterns listed in Fig. 3 for each sentiment pattern. Example 4 details this construction.

Example 4. Let $\mathcal{V}^+$ be a dictionary that contains a set of positive adjectives, $\overline{\mathcal{V}^+}$ be a dictionary that contains the antonyms of each vocabulary $v \in \mathcal{V}^+$. Assume that $(\texttt{good:J}) \in \mathcal{V}^+$, and we have its antonyms $(\texttt{bad:J}), (\texttt{awful:J}), (\texttt{suck:J}) \in \overline{\mathcal{V}^+}$. Thus if we have an instantiated sentiment pattern $p = \langle (\texttt{be:V}) \rangle \Rightarrow \langle (\texttt{good:J}) \rangle$ such that $p \in \mathcal{P}_M^+$ and $p \models \text{VJ}$, then we have the following beliefs generated from

the pattern p with the format $b = [s_\alpha; s_\beta; s_\gamma]$:

$$b_1 = [\langle(\texttt{be:V})\rangle ; \langle(\texttt{good:J})\rangle ; \langle(\texttt{bad:J})\rangle]$$
$$b_2 = [\langle(\texttt{be:V})\rangle ; \langle(\texttt{good:J})\rangle ; \langle(\texttt{awful:J})\rangle]$$
$$b_3 = [\langle(\texttt{be:V})\rangle ; \langle(\texttt{good:J})\rangle ; \langle(\texttt{suck:J})\rangle]$$
$$b_4 = [\langle(\texttt{be:V})\rangle ; \langle(\texttt{good:J})\rangle ; \langle(\texttt{NOT})(\texttt{good:J})\rangle]$$
$$b_5 = [\langle(\texttt{be:V})(\texttt{NOT})\rangle ; \langle(\texttt{bad:J})\rangle ; \langle(\texttt{good:J})\rangle]$$
$$b_6 = [\langle(\texttt{be:V})(\texttt{NOT})\rangle ; \langle(\texttt{awful:J})\rangle ; \langle(\texttt{good:J})\rangle]$$
$$b_7 = [\langle(\texttt{be:V})(\texttt{NOT})\rangle ; \langle(\texttt{suck:J})\rangle ; \langle(\texttt{good:J})\rangle],$$

and if a sentence violates any one of the above beliefs, then this sentence potentially expresses the sentiment opposite to M^+. □

This procedure shown in Algorithm 1 corresponds to such a construction of the belief base.

Algorithm 1: Construction of the belief base from a learning document.

Input : the learning document $\mathcal{D}_L$, the dictionaries $\mathcal{V}$ and $\overline{\mathcal{V}}$, the minimum support value min_supp

Output: the belief base $\mathcal{B}_M$

1 $\mathcal{D}_F = mining\_sequential\_patterns(\mathcal{D}_L, min\_supp)$;
2 $\mathcal{P}_M = \emptyset$;
3 **foreach** $s \in \mathcal{D}_F$ **do**
4   **foreach** $p \in \mathcal{P}$ *(with priority order)* **do**
5   **if** $s \models p$ **then**
6     $p_M = instantiate(p, s)$;
7     $\mathcal{P}_M = \mathcal{P}_M \cup p_M$;

8 $\mathcal{B}_M = \emptyset$;
9 **foreach** $p_M \in \mathcal{P}_M$ **do**
10 $B = generate\_beliefs(p_M, \mathcal{V}^+, \overline{\mathcal{V}^+})$;
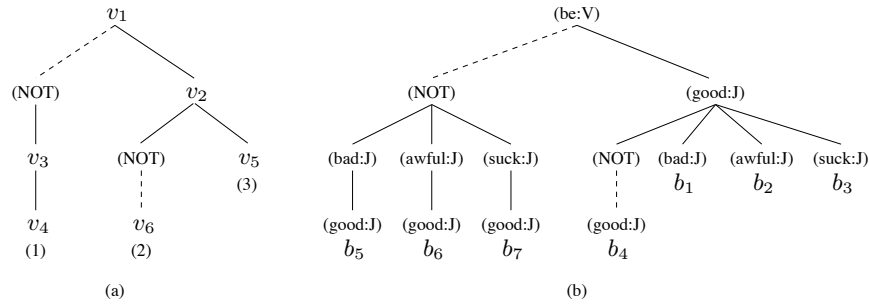11 $\mathcal{B}_M = \mathcal{B}_M \cup B$;
12 **return** $\mathcal{B}_M$;

3.2 Mining Unexpected Sentences

Once the belief base $\mathcal{B}_M$ on sentiment M has been constructed, we can perform the unexpected sentence mining procedure against a document $\mathcal{D}_M$ that has been classified into sentiment M.

We represent the belief base $\mathcal{B}_M$ as the tree structure. Figure 4(a) shows the tree structure of the belief base that consists in 3 beliefs: (1) $[\langle v_1(\texttt{NOT})\rangle ; \langle v_3\rangle ; \langle v_4\rangle]$,

(be:V)

v_1

(NOT) v_2 (NOT) (good:J)

v_3 (NOT) v_5 (bad:J) (awful:J) (suck:J) (NOT) (bad:J) (awful:J) (suck:J)
(3) b_1 b_2 b_3

v_4 v_6 (good:J) (good:J) (good:J) (good:J)
(1) (2) b_5 b_6 b_7 b_4

(a) (b)

Fig. 4. The tree representation of the belief base.

(2) $[\langle v_1 \rangle ; \langle v_2 \rangle ; \langle (\texttt{NOT})v_6 \rangle]$ and (3) $[\langle v_1 \rangle ; \langle v_2 \rangle ; \langle v_5 \rangle]$; Figure 4(b) shows the tree contains 7 beliefs listed in Example 4.

We have two edges in the tree structure shown in Fig. 4. The edge depicted by the discontinued line represents the relations between a vocabulary and (NOT), for example (be:V)(NOT) or (NOT)(good:J); the edge depicted by the solid line represents the presence of a vocabulary in a belief.

Algorithm 2: Extracting unexpected sentences as the opposite sentiment.

Input : the document $\mathcal{D}_M$, the belief base $\mathcal{B}_M$
Output: all unexpected sentences

1 **foreach** $S \in \mathcal{D}$ **do**
2   **foreach** $s_\alpha \in first\_level(\mathcal{B}_M)$ **do**
3   **if** $s_\alpha \sqsubseteq S$ **then**
4   **while** $s_\beta = depth\_first\_traverse(s_\alpha, \mathcal{B}_M)$ **do**
5   **while** $s_\gamma = depth\_first\_traverse(s_\beta, \mathcal{B}_M)$ **do**
6   **if** $s_\alpha \cdot s_\beta \not\sqsubseteq S$ *and* $s_\alpha \cdot s_\gamma \sqsubseteq S$ **then**
7             $output(S)$;

The procedure of extracting unexpected sentences is shown in Algorithm 2. For each sentence $S \in \mathcal{D}_M$, the algorithm steps through the tree represented belief base $\mathcal{B}_M$ for matching S with depth first traverse. For each tree path for standing for a belief $b = \langle s_\alpha; s_\beta; s_\gamma \rangle$, s_α is first matched for improving the performance of the algorithm. If $s_\alpha \sqsubseteq S$, the traverse of belief base let the algorithm match s_β and s_γ within S. If $s_\alpha \cdot s_\beta \not\sqsubseteq S$ and $s_\alpha \cdot s_\gamma \sqsubseteq S$, then the algorithm outputs S as an unexpected sentence, that can be potentially express the opposite sentiment $\overline{M}$.

4 Experiments

The data sets we use for evaluating our approach are the movie-review data[3] introduced in [5]. These data sets are collections of free format movie-review texts labeled with respect to two categories of their overall sentiment polarity, *positive* or *negative*. We combined these reviews into two documents $\mathcal{D}^+$ (containing 1,000 positive reviews) and $\mathcal{D}^-$ (containing 1,000 negative reviews).

In order to maximum avoid the noisy clauses contained in sentences, we consider that a sentence can be terminated by one of ": , ; . ? !", so that finally $\mathcal{D}^+$ contains 75,740 sentences and 21,156 distinct vocabularies; $\mathcal{D}^-$ contains 67,425 sentences and 19,714 distinct vocabularies.

The two dictionaries $\mathcal{V}^+$ and $\mathcal{V}^-$ are generated from $\mathcal{D}^+$ and $\mathcal{D}^-$, by finding most frequent positive/negative adjectives. To not make our experiments too complex, we selected ten most frequent adjectives for each dictionary, listed as Fig. 5. The learning documents $\mathcal{D}^+_L$ (contains 1,678 sentences) and $\mathcal{D}^-_L$ (contains 3,842 sentences) are therefore generated from $\mathcal{D}^+$ and $\mathcal{D}^-$ by gathering the sentences containing at least one adjective from $\mathcal{V}^+$ and $\mathcal{V}^-$.

#	Positive	Frequency	Negative	Frequency
1	good	2146	bad	1414
2	great	882	stupid	214
3	funny	441	poor	152
4	special	282	awful	109
5	perfect	244	silly	97
6	beautiful	202	horrible	71
7	nice	184	suck	65
8	entertaining	179	violent	64
9	wonderful	165	sad	56
10	excellent	146	ugly	44

Fig. 5. The dictionaries $\mathcal{V}^+$ and $\mathcal{V}^-$.

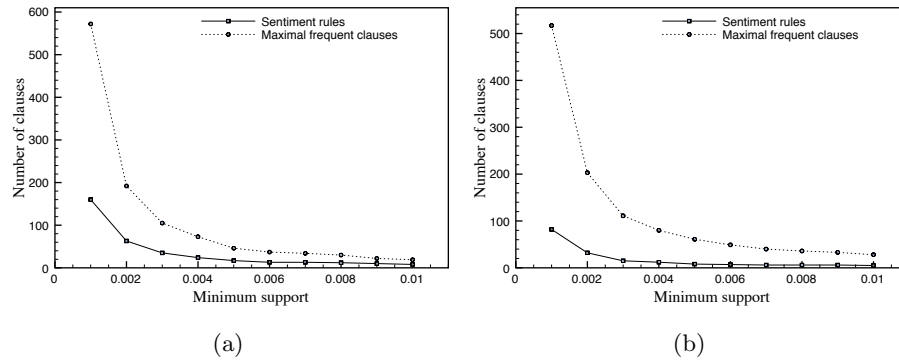The maximal frequent clauses (standing for $\mathcal{D}^+_F$ and $\mathcal{D}^-_F$) and the instantiated sentiment patterns (standing for $\mathcal{P}^+$ and $\mathcal{P}^-$) clauses supporting extracted by the sequential pattern mining algorithm are shown in Fig. 6.

The 10 most frequent instantiated sentiment patterns are listed in Fig. 7. The antonym dictionaries for constructing the belief bases are given by the WordNet project[4]. For respecting the size limit of this paper, we list a small part of the two belief bases in Fig. 8.

In order to analyze the accuracy of our approach, we randomly select a number of beliefs for extracting the sentences that express the sentiment opposite to the documents $\mathcal{D}^+$ and $\mathcal{D}^-$. For instance, as the beliefs listed in Fig. 8, the

---

[3] http://www.cs.cornell.edu/People/pabo/movie-review-data/

[4] http://wordnet.princeton.edu/

Fig. 6. (a) Maximal frequent clauses $\mathcal{D}_F^+$ and instantiated sentiment patterns $\mathcal{P}^+$. (b) Maximal frequent clauses $\mathcal{D}_F^-$ and instantiated sentiment patterns $\mathcal{P}^-$.

Positive Sentiment	Negative Sentiment
$\langle(\texttt{be:V})\rangle \Rightarrow \langle(\texttt{good:J})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{guy:N})\rangle$
$\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{film:N})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$
$\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{movie:N})\rangle$
$\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{performance:N})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{film:N})\rangle$
$\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{movie:N})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{thing:N})\rangle$
$\langle(\texttt{good:J})\rangle \Rightarrow \langle(\texttt{friend:N})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{year:N})\rangle$
$\langle(\texttt{great:J})\rangle \Rightarrow \langle(\texttt{film:N})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{time:N})\rangle$
$\langle(\texttt{great:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$	$\langle(\texttt{bad:J})\rangle \Rightarrow \langle(\texttt{dialogue:N})\rangle$
$\langle(\texttt{special:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$	$\langle(\texttt{stupid:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$
$\langle(\texttt{special:J})\rangle \Rightarrow \langle(\texttt{effect:N})\rangle$	$\langle(\texttt{poor:J})\rangle \Rightarrow \langle(\texttt{be:V})\rangle$

Fig. 7. The 10 most frequent instantiated sentiment patterns.

Belief Base of Positive Sentiment	Belief Base of Negative Sentiment
$[\langle(\texttt{be:V})\rangle\,;\langle(\texttt{good:J})\rangle\,;\langle(\texttt{bad:J})\rangle]$	$[\langle(\texttt{not:R})(\texttt{bad:J})\rangle\,;\emptyset;\langle(\texttt{guy:N})\rangle]$
$[\langle(\texttt{be:V})\rangle\,;\langle(\texttt{good:J})\rangle\,;\langle(\texttt{not:R})(\texttt{good:J})\rangle]$	$[\langle(\texttt{n't:R})(\texttt{bad:J})\rangle\,;\emptyset;\langle(\texttt{guy:N})\rangle]$
$[\langle(\texttt{be:V})\rangle\,;\langle(\texttt{good:J})\rangle\,;\langle(\texttt{n't:R})(\texttt{good:J})\rangle]$	$[\langle(\texttt{bad:J})\rangle\,;\langle(\texttt{be:V})\rangle\,;\langle(\texttt{be:V})(\texttt{not:R})\rangle]$
$[\langle(\texttt{bad:J})\rangle\,;\emptyset;\langle(\texttt{film:N})\rangle]$	$[\langle(\texttt{bad:J})\rangle\,;\langle(\texttt{be:V})\rangle\,;\langle(\texttt{be:V})(\texttt{n't:R})\rangle]$
$[\langle(\texttt{not:R})(\texttt{good:J})\rangle\,;\emptyset;\langle(\texttt{film:N})\rangle]$	$[\langle(\texttt{good:J})\rangle\,;\emptyset;\langle(\texttt{film:N})\rangle]$
$[\langle(\texttt{n't:R})(\texttt{good:J})\rangle\,;\emptyset;\langle(\texttt{film:N})\rangle]$	$[\langle(\texttt{not:R})(\texttt{bad:J})\rangle\,;\emptyset;\langle(\texttt{film:N})\rangle]$
$\ldots\ldots$	$\ldots\ldots$

Fig. 8. The belief bases for mining unexpected sentences.

5 beliefs of positive sentiment produced totally 304 unexpected sentences, and 236 of them express the negative sentiment; the 5 beliefs of negative sentiment produced totally 136 unexpected sentences, and 97 of them express the positive sentiment. Within these beliefs, the average accuracy is about 74.48%.

5 Conclusion

In this paper we present our approach that extracts opposite sentiment as unexpected sentences from classified free text reviews, where the unexpected sentences are discovered with respect to user specified belief bases. We first formalize the problem of finding opposite sentiments, we then propose the algorithms MOSUS, that consists in a procedure for constructing the belief base and a procedure for extracting unexpected sentences. Our experimental results show that the accuracy of the extracted opposite sentiments is in the acceptable range. Our future work includes to develop a belief driven approach for not only finding the opposite sentient, but also helping to the classification process in free text based opinion mining.

References

1. R. Agrawal and R. Srikant. Mining sequential patterns. In *ICDE*, pages 3–14, 1995.
2. K. Dave, S. Lawrence, and D. M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW*, pages 519–528, 2003.
3. A. Esuli and F. Sebastiani. PageRanking wordnet synsets: An application to opinion mining. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL'07), Prague, CZ*, pages 424–431, 2007.
4. M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD*, pages 168–177, 2004.
5. B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *ACL*, pages 271–278, 2004.
6. B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *EMNLP*, pages 79–86, 2002.
7. B. Santorini. Part-of-speech tagging guidelines for the penn treebank project, March 1991.
8. H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, 1994.
9. D. D. Sleator and D. Temperley. Parsing English with a link grammar. In *Third International Workshop on Parsing Technologies*, 1993.
10. P. Turney. Mining the Web for synonyms: PMI–IR versus LSA on TOEFL. In *Proceedings of the Twelfth European Conference on Machine Learning (ECML-2001), LNCS*, pages 491–502, 2001.
11. P. D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424, 2002.