Expert Systems with Applications 39 (2012) 6000–6010

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews

Hanhoon Kang, Seong Joon Yoo *, Dongil Han

Dept. of Computer Engineering, Sejong University, 98 Gunja, Gwangjin, Seoul 143-747, Republic of Korea

ARTICLE INFO

Keywords:
Sentiment analysis
Lexicon
Machine learning
Opinion mining
Restaurant review

ABSTRACT

The existing senti-lexicon does not sufficiently accommodate the sentiment word that is used in the restaurant review. Therefore, this thesis proposes a new senti-lexicon for the sentiment analysis of restaurant reviews. When classifying a review document as a positive sentiment and as a negative sentiment using the supervised learning algorithm, there is a tendency for the positive classification accuracy to appear up to approximately 10% higher than the negative classification accuracy. This creates a problem of decreasing the average accuracy when the accuracies of the two classes are expressed as an average value. In order to mitigate such problem, an improved Naïve Bayes algorithm is proposed. The result of the experiment showed that when this algorithm was used and a unigrams + bigrams was used as the feature, the gap between the positive accuracy and the negative accuracy was narrowed to 3.6% compared to when the original Naïve Bayes was used, and that the 28.5% gap was able to be narrowed compared to when SVM was used. Additionally, the use of this algorithm based on the senti-lexicon showed an accuracy that improved by a maximum of 10.2% in recall and a maximum of 26.2% in precision compared to when SVM was used, and by a maximum of 5.6% in recall and a maximum of 1.9% in precision compared to when Naïve Bayes was used.

1. Introduction

1.1. Section summary

In this thesis, the problem inherent in the previous research is analyzed in order to conduct a sentimental analysis of a restaurant review and the method by which to solve these problems is sought. As its solution, a senti-lexicon associated with a restaurant is constructed and an improved Naïve Bayes algorithm is proposed. In order to prove the effectiveness of the proposed algorithm, the classification result and the accuracy of Naïve Bayes, which is the original classification algorithm, and of SVM are analyzed through a comparative experiment.

1.2. Research background

The number of restaurant reviews continues to increase. Such reviews can be found not only on blogs and the communities but also on local activity information sites and SNS sites such as Yelp.com, Twitter, Facebook and Foursquare. Furthermore, the expansion of smart phones has made it possible to write a review on the spot regardless of the location; thus the amount of reviews continues to increase and the necessity for their integrated search is increasing as well. Many review sites provide both a star score and a review document. However, there are some instances when the star score and the review do not match. For example, it is a case when a star score is high but the review is not good, or when a star score is low but the review is good. Therefore, it may be useful to provide the user with a star score by classifying the sentiment of the review's content as a quantified value through analysis.

1.3. Previous work and problem definitions

In recent research fields the opinion mining field has conducted research that classifies the content of the review in a two-point scale of positive and negative by analyzing it from a qualitative perspective. In an early sentiment analysis, a word that was extracted from the document, as it was in the topic-based document classification approach, was used as the feature. However, in later research a sentiment lexicon was constructed in order to improve the accuracy of the classification, and the vocabulary that was included in the lexicon was used as the feature. The sentiment lexicon represents an index of sentiment words, and it has the polarity information of the relevant word irrespective of whether it carries a positive sentiment or a negative sentiment. Nevertheless, the analysis of the original research shows that a restaurant senti-lexicon has to be constructed in order to appropriately per-

* Corresponding author. Tel.: +82 2 3408 3755.
E-mail address: sjyoo@sejong.ac.kr (S.J. Yoo).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.11.107

form sentiment analysis of the restaurant review when the following characteristics existed.

First, the lexicon that was originally built is either constructed as a general-purpose one or is dependent on a specific domain. Sista and Srinivasan (2004) constructed a lexicon using the General Inquiry Lexicon and the WordNet. Cho and Lee (2006) constructed a lexicon related to the lyrics of songs. Hwang and Ko (2008) constructed a general lexicon that could be used in newspapers, product reviews and movie reviews. These are relatively small lexicons. Turney and Littman (2002) and Kim and Hovy (2004) tried to find a sentiment word through the synonyms and antonyms by giving a seed word that clearly represented the sentiment, such as "good" or "bad". Esuli and Sebastiani (2006) constructed SentiWordNet, which is a general-purpose, mega-size lexicon. In this research, a word that can be applied in the sentimental analysis of a restaurant review is discovered by having built a general sentiment word that can be applied in various domains. However, there are several occasions when the polarity data is not clear. For example, the word "delicious" should be used as a clear positive when evaluating a restaurant, but Esuli and Sebastiani (2006) see "delicious" as positive with a probability of 0.75 when it is used in the sentiment analysis of a restaurant review. On the other hand, Courses and Surveys (2008), Chaumartin (2007), and Fahrni and Klenner (2008) used SentiWordNet. Particularly, Fahrni and Klenner (2008) used the self-developed lexicon and the SentiWordNet together in order to perform a target-specific sentiment analysis. The result of the experiment shows that the classification performance was good when the self-constructed lexicon and the SentiWordNet were used together. This means that it was not able to improve the performance with only the SentiWordNet in a specific domain. Denecke (2009) tried to classify documents such as books and electronics using SentiWordNet. Here, the performance was good in the score and word frequency of SentiWordNet as well as in the machine learning scheme that used the number of the word-specific part of speech as the feature more than the rule-based classification using the score from SentiWordNet. This research showed that the classification performance was not good when only the SentiWordNet was used.

Secondly, Esuli and Sebastiani (2006) and Hwang and Ko (2008) proposed the lexicon of a unigrams pattern made up of one sentiment word such as "good" or "bad." Sista and Srinivasan (2004) considered the unigrams pattern that included negative words such as "not_good" when constructing a lexicon. The problem that can occur here is that a bigrams pattern composed of two words, such as a negative word or an intensive adverb, cannot be found in the lexicon.

Thirdly, the composition ratio of the number of positive words and the number of negative words within a lexicon can have an effect on the classification result. However, the original research did not compare the difference by configuring it in various ways. An evaluation is necessary in order to assess how it is affected by the change in component ratio. In the experiments by Sista and Srinivasan (2004) and Cho and Lee (2006), the number of positive words was more than the number of negative words in a lexicon. As a result, Sista and Srinivasan (2004) showed a classification accuracy of 61.75–84.20%, and Cho and Lee (2006) showed a classification accuracy of 30–83.6%. Hwang and Ko (2008) conducted the experiment while maintaining a similar ratio between the number of positive words and the number of negative words. This showed a classification accuracy of 62.1–77.7%. The research expressed the positive classification accuracy and negative classification accuracy together, and a maximum of 9.6% difference occurred between the two classes.

Fourthly, the result of a sentiment analysis varies according to the composition method of a domain and feature and the type of learning algorithm, but the research that performs a sentiment analysis of a restaurant review by variously combining a feature and a learning algorithm is rare. Sista and Srinivasan (2004) used only unigrams as a feature and classified the movie review as positive or negative by using five algorithms. Hwang and Ko (2008) used only unigrams as a feature, classifying the newspaper, product reviews and movie reviews as positive or negative by using only SVM as the algorithm. Cho and Lee (2006) used the unigrams, bigrams and trigrams as a feature, classifying eight types of sentiments in a person from the lyrics of songs by using only SVM as the algorithm.

Finally, there are occasionally times when the positive classification accuracy and the negative classification accuracy are not accurate in the sentiment analysis. This becomes a factor in lowering the average classification accuracy. Ye, Zhang, and Law (2008) used three kinds of classification algorithms. When Naïve Bayes was used among them, the average classification accuracy of both classes fell to 73.5% with 77.2% positive and 69.9% negative. Similar cases were observed in several sentiment analysis research studies (Dave, Lawrence, & Pennock, 2003; Pang, Lee, & Vaithyanathan, 2002; Ye et al., 2008; Turney, 2002). Accordingly, there is a problem with the decrease in the average classification accuracy when the positive accuracy and negative accuracy are calculated as an average value. Thus there is a need to improve the average classification efficiency by increasing the accuracy of the relatively lower area between the positive and the negative.

1.4. Basic idea of solving the problems

This thesis proposes the following solutions through the problems in the original research.

First, a lexicon that is constructed to facilitate the analysis of a specific domain such as a newspaper, product review or movie review, and the lyrics of songs or a general-purpose lexicon are not sufficient for the sentiment analysis of a restaurant review. Therefore, in this thesis a lexicon that includes the sentiment words associated with a restaurant review is constructed by analyzing the restaurant reviews.

Secondly, the original lexicon was built with a unigrams pattern, but in such a case it is difficult to find a polarity information of the sentiment word with a bigram pattern that has a negative word or an adverb that intensifies the meaning of the sentiment. Thus the unigrams pattern as well as the bigrams pattern, which includes negative words and intensive adverbs, were included in the lexicon in this thesis. Additionally, the experiment was conducted by considering the unigrams + bigrams pattern that combines the unigrams and the bigrams for the purpose of performance comparison.

Thirdly, the classification performance is affected according to the composition ratio of the number of positive words and the number of negative words in a lexicon. It is assumed that the experiment that compares this has not previously been conducted. Assuming that the composition ratio of the number of positive words and the number of negative words can have an effect on the experimental result, this thesis compares the experimental results by forming three kinds of lexicons as in the following: The lexicon that is constructed in this thesis has more positive words than negative words. Accordingly, (1) when there are more positive words than negative words; (2) when there is the same number of positive words as negative words, sorting in the order of descending frequency; and (3) when the number of positive words and the number of negative words are the same, sorting in the order of increasing frequency were considered. As a result, when the improved Naïve Bayes was used, (1) and (2) showed good performance, and (3) showed good performance in SVM. A detailed explanation of this is provided in Section 3.

Fourthly, the experiment was conducted by using the three types such as the unigrams, bigrams and unigrams + bigrams as a feature in order to find out which combination works in a restaurant review, and by applying SVM, Naïve Bayes and the improved Naïve Bayes, which is proposed in this thesis, as the sentiment analysis algorithm.

Lastly, the positive classification accuracy and the negative classification accuracy did not achieve similar levels but instead showed differences relative to the original research. Ultimately, such results decrease the average classification accuracy. Even with the restaurant review that is used as a subject in this thesis, when sentiment analysis is performed using an existing supervised learning algorithm such as SVM and Naïve Bayes, the average classification accuracy decreased by showing a difference between the accuracy of the positive classification and the accuracy of the negative classification. Thus two types of improved Naïve Bayes algorithms are proposed in this thesis in order to minimize the gap between the accuracies of a positive classification and a negative classification, and to improve the average classification accuracy.

The proposed two algorithms improve the likelihood by referring to the pattern included in a senti-lexicon. The first Improved Naïve Bayes algorithm narrowed the gap between a positive classification accuracy and a negative classification accuracy to 3%, compared to when Naïve Bayes was used. The difference narrowed 28.2% compared to when SVM was used. When the second Improved Naïve Bayes algorithm was used, the difference narrowed 3.6% compared to when Naïve Bayes was used, and it narrowed to 28.5% compared to SVM. The more significant effect of using the Improved Naïve Bayes can be seen in the improvement of classification accuracy. This means that the use of the second Improved Naïve Bayes algorithm showed an accuracy that improved by a maximum of 5.6% in recall and a maximum of 1.9% in precision compared to when Naïve Bayes was used, along with an improvement accuracy of up to 10.2% in recall and a maximum of 26.2% in precision compared to when SVM was used. The related research is described in Section 2 of this thesis. Section 3 analyzes the problems in the original research that constructed the lexicon and explains the method of constructing a restaurant senti-lexicon as its solution. Section 4 analyzes the problems in the original research associated with the sentiment analysis, and proposes solutions. Section 5 explains the new classification algorithm that is proposed in this thesis. Section 6 analyzes the effectiveness of the proposed method through an experiment. Finally, Section 7 is a conclusion.

2. Related works

This section describes the major research on the positive and negative classification research of the document level (Dave et al., 2003; Hu & Li, 2010; Hwang & Ko, 2008; Kang, Yoo, & Han, 2009; Kim, Jung, & Mayeng, 2008; Miao, Li, & Dai, 2009; Pang et al., 2002; Prabowo & Thelwall, 2009; Tan & Zhang, 2008; Turney, 2002; Ye et al., 2008; Zhu, Xu, & Wang, 2010), the positive and negative classification research of the feature-level (Ding, Liu, & Yu, 2008; Hu & Liu, 2004; Kim & Hovy, 2004; Miao, Li, & Dai, 2008; Popescu & Etzioni, 2005; Su, Xiang, Wang, Sun, & Yu, 2006; Yang, Kim, & Lee, 2010; Yang, Myung, & Lee, 2009), and finally the sentiment analysis research based on the (Cho & Lee, 2006; Esuli & Sebastiani, 2006; Fahrni & Klenner, 2008; Myung, Lee, & Lee, 2008; Sista & Srinivasan, 2004).

Pang et al. (2002), Ye et al. (2008) and Kang et al. (2009) classified a review document as positive or negative by applying a traditional topic-based document classification method. Dave et al. (2003) determined positive or negative stochastically by using a score function formula. Turney (2002) classified positive or negative at the document level using PMI-IR. This thesis calculates the Semantic Orientation (SO) of multiple phrases that satisfy a specific phrase pattern with a PMI-IR formula. Afterwards, it is classified as positive if the sum of SO is a positive number or as negative when it is a negative number on each phrase. Kim et al. (2008) calculated the Document Sentiment Value (DSV) using a self-defined formula. Afterwards, it classified as positive or negative according to the value of DSV. Hwang and Ko (2008) manually constructed part of the sentiment word, and by forming it into a feature vector they classified as positive or negative with a supervised learning algorithm. Miao et al. (2009) measured the ranking by analyzing the quality of an opinion with a study on the sentiment mining and retrieval system. Additionally, an opinion is classified internally as positive or negative using the Naïve Bayes algorithm. Tan and Zhang (2008) classified Chinese documents as positive and negative. Here, the document frequency (DF), CHI statistical measures, mutual information and the information gain were used as a feature selection method, and the Centroid classifier, KNN, Naïve Bayes, WinNOW classifier and SVM classifier were used as machine learning methods. Prabowo and Thelwall (2009) performed the classification of movie reviews, product reviews and Myspace comments by combining the rule-based classification and supervised learning algorithm. As a result, the hybrid classification showed an improvement in accuracy. Zhu et al. (2010) classified the Cornell movie review as positive or negative by using an ANNs-based individual model (i-model), which showed a better classification result than either the Support Vector Machine (SVM) or the Hidden Markov Model (HMM). Ye, Shi, and Li (2006) classified the positive and the negative by applying the PMI-IR method proposed by Turney (2002) to a Chinese document. Denecke (2009) performed a rule-based classification and a machine learning classification. A rule-based classification brings a positive score and a negative score of a word that is included in a document from the SentiWordNet. At this time an average value by class must be obtained, because multiple results are obtained. This is a method that classifies toward a higher value between a positive average value and a negative average value. A machine learning classification uses a Simple Logistical Classifier by using an average score and a frequency of word by class as well as a number of parts of speech of a word as a feature. This classifier is based on a logistical regression model and is in the WEKA Library (Witten & Frank, 2005, chap. 6).

Feature-level classification research has been conducted along with sentiment word extraction research. Hu and Liu (2004) assumed a noun as a feature and found a feature through an association rule. Subsequently, a close adjective of a feature was extracted as a sentiment word. Yang et al. (2010) organized and summarized the satisfaction by the product's attribute with grading. Kim and Hovy (2004) found a feature by applying a named entity tagger and found a sentiment word by using WordNet. In an early stage, multiple sentiment words are given as seed words, and a lexicon is constructed by expanding the synonyms and antonyms of each seed word. Popescu and Etzioni (2005) found a feature by using the same method as Hu and Liu (2004) and, using it as the center, they found a sentiment word with a rule that could express the sentiment word. Su et al. (2006) and Fahrni and Klenner (2008) manually calculated to which word the extracted feature was related by using a PMI-IR formula and treating a word most closely related to a specific feature as the sentiment word. Ding et al. (2008) considered the context when finding the polarity information of a sentiment word in a feature unit. Consideration of the context is as follows: For example, the sentiment word "small" in a portable device is positive, but "big" is positive in a device such as a monitor or a TV.

There is the SentiWordNet (Esuli & Sebastiani, 2006) as a research study that constructed a lexicon in English. This is constructed for the sentiment analysis of various fields by expanding

WordNet, and as such it is of general purpose. Yet another research study, Sista and Srinivasan (2004), constructed a lexicon by using the General Inquirer Lexicon and the WordNet database. In this research a classification experiment of movie reviews was also conducted through the use of a lexicon. In this research a supervised learning algorithm such as Naïve Bayes or SVM was used for classification. Fahrni and Klenner (2008) constructed a self-made lexicon with a two-stage model, and the classification performance was good when the self-constructed lexicon and SentiWordNet were used simultaneously. Cho and Lee (2006) manually constructed a lexicon in order to find eight kinds of basic emotions in a person from the lyrics of songs, and performed a classification experiment using a supervised learning algorithm. Myung et al. (2008) analyzed a product review with a syntax analyzer and extracted a sentiment word candidate from a predicate. Among the sentiment word candidates, a method that selects a word with an occurrence frequency in the overall document higher than the threshold as the final sentiment word was proposed. However, because in reality there may be occasions when even an arbitrary word with a value less than the threshold is the sentiment word, such a word cannot be selected as the sentiment word.

3. Construction of senti-lexicon for restaurant reviews

In this section the first, second and third problems associated with the construction of the lexicon among the five problems mentioned in the introduction are described in detail, and the methods for their improvement are explained. Esuli and Sebastiani (2006) developed the lexicon SentiWordNet (http://sentiwordnet.isti.cnr.it/ (last accessed Nov. 22, 2010)) with a general purpose. Table 1 organizes the output result after the word "delicious", which can be commonly used in restaurant reviews, is queried in the system. Based on the result, the difficulty in determining the polarity of the word that can be used to review a restaurant can be summarized in three points. First, the program does not know which word's sentiment score among the three results must be applied. Secondly, although the positive score of No. 1, which can be used to review the food's taste, is high relative to the objective score and the negative score, "delicious" is given only 75% probability in evaluating the taste of food even though it is close to 100% positive. Thirdly, a sentiment word that is made up of two words such as "not delicious" cannot be found in SentiWordNet. This is because SentiWordNet only expresses the sentiment score for one word.

Even though "greasy", which can be used to evaluate the food's taste besides "delicious", should be used as a negative word of high probability, its negative score is set as a low probability of 0.25 in SentiWordNet. Moreover, the negativity score of "greasy", which is also used to mean "greasy coveralls", is set as a probability of 0.875, but there is a difficulty in determining the polarity of the word that evaluates the taste of food.

This thesis collected restaurant reviews in order to improve the problems in the existing research, and a restaurant senti-lexicon was developed after manually analyzing the collected data. This lexicon also includes the bigram pattern, which is made up of two words that are not handled by Esuli and Sebastiani (2006). This means it includes both the patterns made up of one word, such as "good" and "delicious", and the patterns that are made up of two words, such as "not delicious" and "very delicious." The restaurant senti-lexicon (http://irlab.sejong.ac.kr/res-sentiwordnet/) that is developed in this thesis is constructed with the polarity information such as positive or negative in the sentiment word that is associated with the restaurant review. The process of its construction is as follows:

(1) Reviews associated with a restaurant were collected by developing a wrapper-based Web crawler.
(2) A part of speech was tagged by eliminating the unnecessary characters from the collected reviews, such as a number, symbol and HTML tag, and by using a POS tagger.
(3) The unigrams, bigrams and sentiment patterns were found by a manual process and were added to the lexicon by determining the positivity and negativity of the applicable pattern.

The senti-lexicon L is made up of a set as shown in (1).

$$L = \{TYPE, TARGET, PATTEN, POLARITY\} \tag{1}$$

The values that correspond to *TYPE* are unigrams or bigrams. The unigrams in TYPE refer to a pattern made up of one word. The examples of such pattern are "cheap", "delicious" and "good". The bigrams is a pattern type made up of two words. The examples of such pattern include "very delicious", where the first word is an accentuating adverb (Z) that stresses the meaning of "delicious"; and "not delicious", where a negative word is attached and gives it the opposite meaning. A *TARGET* represents a subject that expresses an emotion. This represents an evaluative attribute such as the food's taste, facility, mood and price that can be felt when visiting a restaurant. When *TARGET* is not shown explicitly, it is expressed as common. A *PATTERN* expresses an emotive word pattern. A part of speech is also expressed at this time. It is made up of one word (e.g. Nos. 1 and 2) or two words (e.g. Nos. 3 and 4) according to a *TYPE*.The parts of speech of words that are considered in this thesis are adjectives, verbs and nouns. Additionally, even with a same part of speech, a parameter whereby the extraction should be made by each part of speech was selected. Because most of the words paired with an adjective as the part of speech express emotions, the decision was made to extract all adjective words. It was decided that words in verbs and nouns that express emotions would be extracted. For example, "recommend" is a verb, but a word such as "recommendation" or "kindness" is expressed as a verb or an adjective in the form of a noun in a noun. Additionally, a word such as "eat", which carries little emotive meaning, is eliminated and a pattern that expresses an emotion through an adverbial phrase such as "eat deliciously" is built into a lexicon. A *POLARITY* is expressed as positive or negative. Table 2 shows an example of a lexicon that is constructed in this thesis. The number of patterns that are constructed here is 436 positive unigrams patterns and 236 negative unigrams patterns. With 1159 positives and 656 negatives, the bigrams pattern is constructed with a total of 2487 patterns. Numbers 1 and 2 in Table 2 show an example of a unigram pattern that is made up of an adjective (V), and No. 3 shows an example of a bigram pattern that is made up of an adverb (Z) and an adjective (V). Number 4 also shows an example of a

Table 1
Result when "delicious" is queried in SentiWordNet 3.0.

No.	Meaning	Part-of-Speech	Positivity score	Objectivity score	Negativity score
1	greatly pleasing or entertaining	adjective	0.75	0.25	0
2	extremely pleasing to the sense of taste	adjective	0.75	0	0.25
3	variety of sweet eating apples	noun	0	1	0

Table 2
Example of restaurant senti-lexicon.

No.	TYPE	TARGET	PATTERN	POLARITY
1	unigrams	Common	good_V	Positive
2	unigrams	Food material	fresh_V	Positive
3	bigrams	Price	very_Z expensive_V	Negative
4	bigrams	Food tasty	not_NOT delicious_V	Negative

Table 3
An example of a unigrams pattern.

Positive pattern	Frequency	Negative pattern	Frequency
good	14,887	greasy	1367
delicious	13,299	sad	2441
nice	4034	hate	116
recommend	1367	...	
cheap	123		
...			

Table 4
An example of a bigrams pattern.

Positive pattern	Frequency	Negative pattern	Frequency
very excellent	1178	not good	699
really good	2640	very expensive	332
not bad	831	really hate	19
very delicious	68	...	
too inexpensive	31		
...			

bigram pattern, being a pattern made up of an adjective (V) and a negative word (NOT) but whose polarity has the opposite meaning.

As explained above, the lexicon constructed in this thesis has a higher number of positive patterns than negative patterns. Also, each pattern made up of a lexicon generally has a higher frequency of positive patterns. Tables 3 and 4 show those characteristics. The frequency of a pattern and the composition of an unbalanced lexicon are considered to show a biased result in the classification results. Therefore, in this thesis a dictionary is composed in three types.

4. Sentiment analysis of restaurant reviews using the SVM and the Naïve Bayes algorithm

In the existing research it is rare to find a case that compares performance by performing the sentiment analysis of a restaurant review by applying more than two features and more than two classification algorithms. This section describes a method of performing the sentiment analysis of a restaurant review by using both SVM and Naïve Bayes, doing so by applying a combination of more than two features. The feature used in this thesis is the same as the pattern constructed in the lexicon. That means a unigrams feature is made up of one word such as "delicious_V" and has a Part-of-speech information such as "V". A bigrams feature is made up of two words such as "very_Z delicious_V", and it has the part-of-speech information for each word. Finally, the unigrams + bigrams feature uses the combination of the unigrams and the bigrams features in an experiment. The Support Vector Machine (Han & Kamber, 2005, chap. 6; Witten & Frank, 2005, chap. 6; Liu, 2006) that was used in this thesis for sentiment analysis classifies positive and negative by linearly separating the different types of input vectors. The circle and the square shown in Figs. 1 and 2 represent an input vector, while the circle represents positive and the square represents negative. In a process of creating the first training model in SVM, we find an optimal hyperplane for separating the input vectors between the two classes as shown in Fig. 1. At this time, although various forms of hyperplanes can be found, a hyperplane is searched while considering the maximum margin. Fig. 2 is an example of a hyperplane that was found with consideration for the maximum margin. Here, a filled circle and a filled square are called support vectors. These have all the information that can classify positive and negative, and the remaining vectors are discarded.

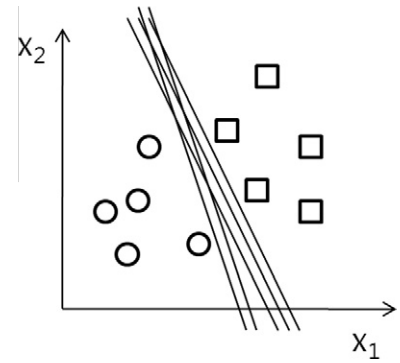

Fig. 1. Optimal hyperplane search.

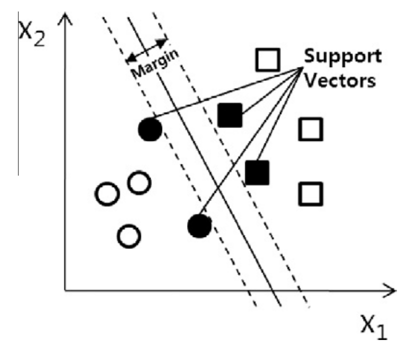

Fig. 2. Optimal hyperplane decision.

The decision function used to determine the optimal hyperplane in Fig. 1 is same as (1)

$$f(x) = \sum_{i=1}^{n} a_i y_i (x_i, x) + b \tag{1}$$

In formula (1):

$f(x)$ is a decision function for determining the optimal hyperplane when an input vector x is given.
a: This is a support vector and it corresponds to a filled circle and a filled square in Fig. 2 ($a > 0$).
y: This signifies a class, and it has a value of 1 or -1. In Fig. 2, circle signifies 1, and square signifies -1.
x: This signifies an input vector. In Fig. 2 it corresponds to a circle or a square.
b: This signifies a bias. It signifies the distance from the origin to the hyperplane.

Additionally, when a linear separation is not possible in SVM, it is made to be linearly separable by converting the N-dimensional vector to a higher dimensionality by using a kernel function. Formula (2) represents a kernel function K and a decision function for conversion to a higher dimensionality when a linear separation is not possible

Table 5
An example of input vector.

	f_1	f_2	...	f_n
d_1	w_{11}	w_{12}		w_{1n}
d_2	w_{21}			
d_m	w_{m1}			

$$K(x_i, x_j) \equiv \phi(x_i)^T \cdot \phi(x_j)$$
$$f(x) = \sum_{i=1}^{n} a_i y_i K(x_i, x_j) + b \tag{2}$$

In formula (2):

$K(x_i,x_j)$: This is a kernel function, and it converts into a higher dimensionality when a linear separation is not possible. Here the input vector x_i is mapped to a higher dimensional space by the function ϕ.

On the other hand, the input vector used as the input value of SVM is shown in Table 5. This is a document-feature matrix composed of weight.

Here the feature f_i signifies a sentiment pattern that is included in document d_j, and w_{ij} signifies weight. When a classification experiment is carried out without applying the lexicon in SVM, the pattern (e.g. unigrams, bigrams) extracted from the review document is used as a feature, just as in the traditional topic based document approach. The feature number for this is shown in Table 6. The feature is extracted based on the frequency, and the weight is set as 1.

However, when a classification experiment is conducted using the lexicon in SVM, a general pattern as configured in Table 6 and a pattern included in the lexicon were mixed. Nevertheless, the feature number in Table 6. stayed the same. For example, the count of the positive patterns in the unigrams is 436, but when a lexicon is used in SVM, the positive pattern in the unigrams is made up of 2064 general patterns and 436 lexicon patterns. Here, the weight of the general pattern is given as 0.5 and that of the lexicon pattern is given as 1. This is to give a high weight value to the sentiment pattern included in the lexicon.

The other algorithm used in this thesis, Naïve Bayes, performs the sentiment analysis based on the conditional probability as shown in formula (3)

$$Class(d_i) = \arg\max P(c_j) \prod_{i=1}^{d} P(p_i|c_j) \tag{3}$$
$$P(p_i|c_j) = \frac{|c_j| + 1}{|V| + \sum_{i=1}^{|d|}\sum_{k=1}^{|C|} f(p_{ik})} \tag{4}$$
$$P(c_j) = \frac{|c_j|}{\sum_{k=1}^{|C|} |c_k|} \tag{5}$$

In formula (3):

$Class(d_i)$: This is a function that determines the class (ex. positive or negative) of the document d_i.
$P(c_j)$: Calculates the probability of class c_j.
$P(p_i|c_j)$: Calculates the probability that the pattern (ex. unigrams, bigrams) p_i belongs to class c_j.

Table 6
The feature number used in the SVM classifier.

	Positive pattern	Negative pattern
unigrams	2500	2500
bigrams	10,000	10,000
unigrams + bigrams	10,000 (uni.: 2500/bi.: 7500)	10,000 (uni.: 2500/bi.: 7500)

In formula (4):

$|c_j|$: Refers to the count of patterns that are included in the class c_j. The reason that 1 is added here is to prevent $|c_j|$ as the Laplace estimator from effecting the entire formula when it becomes 0.
$|V|$: Refers to the count of the patterns that are included in the entire document. This means that $|c_{j=positive}| + |c_{j=negative}|$.
$\sum_{i=1}^{|d|}\sum_{k=1}^{|C|} f(p_{ik})$: Calculates the frequency of the pattern that is included in the entire document.

Formula (5):

$|c_j|$: Has the same definition as the formula used in formula (4).
$\sum_{k=1}^{|C|} |c_k|$: Calculates the count of the patterns in the entire document (entire class).

On the other hand, Naïve Bayes does not require an input vector as the input value for the sentiment analysis, as it does with SVM, and it is based on the probability value of each feature.

5. Improved Naïve Bayes algorithm

In this section two improved Naïve Bayes algorithms are proposed as methods that can improve the imbalance problem of the positive classification accuracy and the negative classification accuracy, which is described in the introduction as the fifth problem. With this algorithm the gap between the positive classification accuracy and the negative classification accuracy can be minimized while increasing the accuracy.

5.1. INB-1 algorithm

(Definition) INB-1 is defined as a formula that is made up of the following formulas (6) and (7). Formula (6) is formula (3) with the addition of R_1

$$Class(d_i) = \arg\max R_1(p_{ij}) P(c_j) \prod_{i=1}^{d} P(p_i|c_j) \tag{6}$$
$$R_1(p_{ij}) = \frac{\sum_{p_{ij}\in L_j}^{|L|} C(p_{ij})}{\sum_{p_{ij}\in L}^{|L|} C(p_{ij})} \tag{7}$$

In formula (6):

$R_1(p_{ij})$: The ratio of pattern p_{ij} that applies to class j is calculated from the senti-lexicon when the likelihood of class j is calculated. This is multiplied with the final likelihood of the formula (3) after the positive probability and the negative probability of the document d are again calculated through the senti-lexicon.

In formula (7):

L: Refers to a senti-lexicon.
$|L|$: Refers to the number of patterns that are included the senti-lexicon.
$\sum_{p_{ij}\in L_j}^{|L|} C(p_{ij})$: If the pattern p_i is included in the class j in the lexicon, its number is counted.
$\sum_{p_{ij}\in L}^{|L|} C(p_{ij})$: If p_i is included in the lexicon with the entire class as the target, its number is counted.
$C(p_{ij})$: The number of the pattern is counted.

(Example) Calculation of R_1

Let us assume that there is a document set d as shown in formula (8). In d, let us assume that p_1, p_2, p_3 are portions of the positive pattern in the lexicon and p_4, p_5 are portions of the negative pattern in the lexicon. In such an instance the ratio of the positive pattern, $R_{1(positive)}$, is 3/5, and the ratio of the negative pattern, $R_{1(negative)}$, is 2/5 in the document set d.

$$d = \{p_1, p_2, p_3, p_4, p_5\} \tag{8}$$

5.2. INB-2 algorithm

(Definition) INB-2 defines formula (4) as formula (9). In INB-2, the probability values of each pattern appearing in the document d are made to be clearly classifiable as positive or negative

$$P(p_i|c_j)R_2(p_{ij}) \tag{9}$$

$$R_2(p_{ij}) = \begin{cases} \text{if } p_{ij} \in L_j & \alpha = 0.9999 \\ \text{else} & \beta = 0.0001 \end{cases} \tag{10}$$

In formula (9):

$P(p_i|c_j)$: Has the same meaning as formula (4).

In formula (10):

$R_2(p_{ij})$: When calculating the likelihood of the class j, if the pattern p_{ij} is included in the class j in the senti-lexicon, it is calculated as 0.999, and if not it is calculated as 0.0001.

(Example) Calculation of R_2

Among the words included in the document d of (8), if p_1, p_2, p_3 are the positive patterns and p_4, p_5 are the negative patterns, formula (9) can be expressed as formula (11) when the document d is trying to obtain the probability (c_0) of being positive

$$P(p_1|c_0)\alpha * P(p_2|c_0)\alpha * P(p_3|c_0)\alpha * P(p_4|c_0)\beta * P(p_5|c_0)\beta \tag{11}$$

On the other hand, formula (9) can be expressed as formula (12) when the probability (c_1) of being negative is obtained

$$P(p_1|c_0)\beta * P(p_2|c_0)\beta * P(p_3|c_0)\alpha * P(p_4|c_0)\alpha * P(p_5|c_0)\alpha \tag{12}$$

6. Analysis of the improved Naïve Bayes algorithm

6.1. Classification using the Naïve Bayes algorithm

When a restaurant review is given based on the formula proposed in Sections 4 and 5, the sentiment analysis process based on the Naïve Bayes algorithm (referred to as NB1), INB-1 and INB-2 is described. A training phase is needed first in order to classify the restaurant review. In this step a training model, such as Table 7, is constructed. Here the unigrams, bigrams and the unigrams + bigrams pattern, as well as their probability values that were extracted from the training data, are included. The probability values are calculated by the formulas (4) and (5). The pattern in Table 7. is an emotive word pattern, and it carries the part-of-speech information. $P(p_i|c_{positive})$ signifies the probability that pattern p_i would be included in the positive training data, and $P(p_i|c_{negative})$ signifies the probability that pattern p_i is included in the negative training data. On the other hand, it assumes $P(c_{positive})$ = 0.5, and $P(c_{negative})$ = 0.5.

Table 7
An example of a training model.

Pattern	$P(p_i\|c_{positive})$	$P(p_i\|c_{negative})$
delicious_V	0.6	0.4
cheap_V	0.4	0.6
noisy_V	0.6	0.4
...		

A classification is performed on the restaurant review that is given after a training module is constructed. For example, if the restaurant review states that "foods are delicious and cheap, but the mood is noisy", then "food", "are", "but", "mood" and "is" do not represent emotion, thus they are eliminated by the preprocessing process and the part-of-speech tagging process. The pattern that is ultimately extracted is "delicious_V", "cheap_V" and "noisy_V". By referring to the probability values on such pattern from the training model, the positive likelihood is calculated as shown in formula (13) and the negative likelihood is calculated as shown in formula (14). The classification result does not belong to either the positive side or the negative side. That is because the likelihoods of two classes are identical

$$\begin{aligned} likelihood_{positive} &= P(\text{'}delicious\_V\text{'}|c_{positive})P(\text{'}cheap\_V\text{'}|c_{positive}) \\ &\quad \times P(\text{'}noisy\_V\text{'}|c_{positive})P(c_{positive}) \\ &= 0.6 * 0.4 * 0.6 * 0.5 = 0.072 \end{aligned} \tag{13}$$

$$\begin{aligned} likelihood_{negative} &= P(\text{'}delicious\_V\text{'}|c_{negative})P(\text{'}cheap\_V\text{'}|c_{negative}) \\ &\quad \times P(\text{'}noisy_{-V}\text{'}|c_{negative})P(c_{negative}) \\ &= 0.4 * 0.6 * 0.4 * 0.5 = 0.072 \end{aligned} \tag{14}$$

The problem that shows up in NB1 is that even though "cheap_V" has a positive meaning, when the overall content of the document is negative there is an increased possibility that this word alone can be included in the set of patterns that determine a negative sentiment. Similarly, the "noisy_V" can be included in the set of patterns that determine a neutral sentiment, even though it is negative.

6.2. Classification using the INB-1 algorithm

The improved Naïve Bayes that is proposed in this thesis can solve the problem that occurs in NB1 by referencing the polarity data of the emotion pattern from the senti-lexicon. In the senti-lexicon, if "delicious_V" and "cheap_V" are shown as positive and "noisy_V" is shown as negative, the likelihood is readjusted in INB-1 as shown in (15), (16). R_1, which is used in the formulas (15) and (16), represents the probability that a pattern is included in the senti-lexicon. In a given review, the number of patterns that has emotions is 3, and among them the positive pattern is 2 because they are "delicious_V" and "cheap_V". Negative pattern is 1 because it is "noisy_V." Thus, R_1 in positive is (2/3 = 0.67) and R_1 in negative is (1/3 = 0.33). Because the positive likelihood is higher than the negative likelihood in INB-1, it is classified as positive

$$\begin{aligned} likelihood_{positive} &= R_1 P(\text{'}delicious\_V\text{'}|c_{positive})P(\text{'}cheap\_V\text{'}|c_{positive}) \\ &\quad \times P(\text{'}noisy\_V\text{'}|c_{positive})P(c_{positive}) \\ &= (2/3) * 0.6 * 0.4 * 0.6 * 0.5 = 0.048 \end{aligned} \tag{15}$$

$$\begin{aligned} likelihood_{negative} &= R_1 P(\text{'}delicious\_V\text{'}|c_{negative})P(\text{'}cheap\_V\text{'}|c_{negative}) \\ &\quad \times P(\text{'}noisy\_V\text{'}|c_{negative})P(c_{negaitive}) \\ &= (1/3) * 0.4 * 0.6 * 0.4 * 0.5 = 0.024 \end{aligned} \tag{16}$$

6.3. Classification using the INB-2 Algorithm

The likelihood is obtained through the calculation as shown in formula (9) and formula (10) in INB-2, through whether each pattern is included in the senti-lexicon. The formulas (17) and (18) show examples of calculating the likelihood in INB-2. Unlike INB-1, the likelihood is readjusted by controlling the probability value of each pattern in INB-2. Because the positive likelihood is also higher than the negative likelihood in INB-2, it is classified as positive

$$\begin{aligned} likelihood_{positive} &= P(\text{'}delicious\_V\text{'}|c_{positive})R_2 P(\text{'}cheap\_V\text{'}|c_{positive})R_2 \\ &= P(\text{'}noisy\_V\text{'}|c_{positive})R_2 P(c_{positive}) = (0.6 * 0.9999) \\ &\quad * (0.4 * 0.9999) * (0.6 * 0.0001) * 0.5 = 7.199e-06 \end{aligned} \tag{17}$$

$$\begin{aligned} likelihood_{negative} &= P(\text{'}delicious\_V\text{'}|c_{negative})R_2 P(\text{'}cheap\_V\text{'}|c_{negative})R_2 \\ &= P(\text{'}noisy\_V\text{'}|c_{negative})R_2 P(c_{negative}) = (0.4 * 0.0001) \\ &\quad * (0.6 * 0.0001) * (0.4 * 0.9999) * 0.5 = 7.199e-10 \end{aligned} \tag{18}$$

INB-1 and INB-2 readjust the likelihood by using the senti-lexicon data in order to resolve the problem that can occur in NB1. In the above example the readjustment of the likelihood shows only the possibility of improvement in the classification performance because it is based on one review. However, when the likelihood is readjusted on multiple reviews, the gap between the positive classification accuracy and the negative classification accuracy can be narrowed. This is shown in the experimental result in Section 7.

7. Experiment and result

In this section the data for the experiment is collected and analyzed, and the manual method of classifying the positive and the negative is explained. At the same time, the experiment proposed in the previous section is conducted and its result is analyzed.

7.1. Restaurant review documents

Approximately 70,000 review documents were collected from the restaurant search sites. The reviews collected in this thesis also include star score. Additionally, the title, author, registration date and contact are included. The supervised learning algorithm requires the training data that is already classified into positive and negative. Accordingly, the positives and negatives of the collected reviews must be manually classified before the sentiment analysis is performed. Among the collected documents, the documents with the star score of 1 were handled as negative and documents with the star score of 5 were handled as positive. However, as described in the introduction, there can be an inconsistency in the content of the review document and its star score. Consequently, it was re-classified manually by referring to the corresponding review's star score. When the characteristics of the reviews that were collected in this thesis were observed, the document with the star score of 1 that was handled as negative was nearly always expressed with negative words. In the document with the star score of 5 that was handled as positive had negative words, but there were relatively more positive words. However, the portion of the documents with the star score of 5 had the positive and negative words appear approximately the same number of times, and these were excluded from the experimental data. Such selected documents totaled 5700 positive documents and 5700 negative documents. Because the number of documents with the star score of 1 among the collected 70,000 documents was 5700, the number of the positive documents was also set as 5700 in order to maintain the ratio of positive documents to negative documents. This data is different from the data used to construct the lexicon.

7.2. Experimental setting

The experimental procedure is divided into a training phase and a classification phase, as shown in Fig. 3. The training phase is a process that builds a training model, and a new restaurant review is classified based on the training model in the classification phase. In each procedure a restaurant review that has been manually sorted in the previous section is received as input. The preprocessing step eliminates the special symbols, HTML tags and other unnecessary characters that are included in each restaurant review. Afterwards, a part of speech for each word from the refined review is tagged. In the feature extraction step a unigrams pattern made up of one word for extracting a sentimental pattern, a bigrams made up of two words, and an unigrams + bigrams pattern combining a unigrams pattern and a bigrams pattern is extracted. The pattern extracted in a training step generates a training model according to each algorithm. For example, a model is constructed by calculating the probability value based on the frequency of the extracted pattern in Naïve Bayes, and a model is built by finding the optimal hyperplane in order to divide the two classes in SVM as shown in Fig. 1. NB1, INB-1, INB-2 were directly implemented, and SVM used a LIBSVM Package (www.csie.ntu.edu.tw/~cjlin/libsvm/ (last accessed November 22, 2010)).

A classification is performed by using the extracted feature and the constructed model in the classification phase. The experimental data was verified by the 10-fold cross validation method on 5700 positive training data and 5700 negative training data that were previously selected. A lexicon was not applied when NB1 was used, and a feature used unigrams, bigrams, and unigrams + bigrams. INB-1 and INB-2 were used when a lexicon was applied; the lexicon used D1, D2 and D3, as explained in Table 8; and a feature used three kinds that were identical to NB1. The experiment was conducted by making a total of 21 kinds of cases with such combinations. The lexicon in SVM also used D1, D2 and D3, as well as DX, as described in Table 8, and the feature used unigrams, bigrams, and unigrams + bigrams. The experiment was conducted by making a total of 12 kinds of cases with such combinations. Here, DX did not apply a lexicon.

7.3. Result

The performance evaluation of the experiment used the Recall and Precision (Kang & Yoo, 2007). The experimental case that is shown in Fig. 4 through Fig. 7 consists of [Classifier]-[Lexicon]-

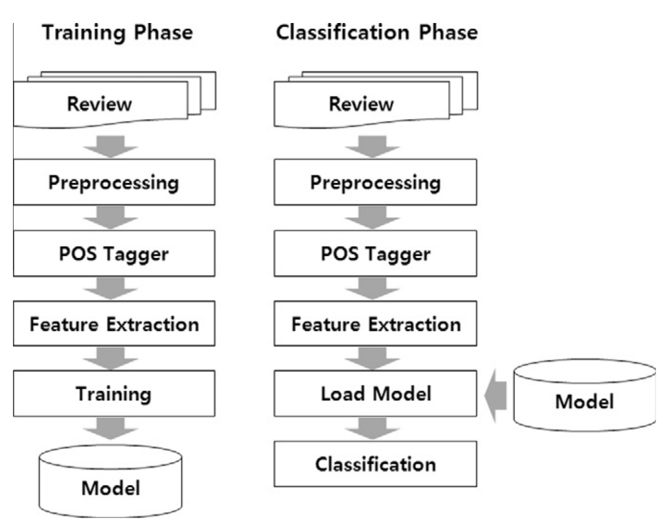

Fig. 3. The phase of training and classification.

Table 8
Type of lexicon.

Type	Descriptions
D1	The number of the positive pattern is higher than the number of the negative pattern (unigrams) positive pattern: 436, negative pattern: 236 (bigrams) positive pattern: 1159, negative pattern: 656
D2	The number of the positive pattern and the number of the negative pattern are maintained as identical (unigrams) positive pattern: 236, negative pattern: 236 (bigrams) positive pattern: 656, negative pattern: 656 (But, just the number of negative pattern is extracted by sorting the frequency of the positive pattern in a descending order)
D3	The number of the positive pattern and the number of the negative pattern are maintained as identical (unigrams) positive pattern: 236, negative pattern: 236 (bigrams) positive pattern: 656, negative pattern: 656 (But, just the number of negative pattern is extracted by sorting the frequency of the positive pattern in an ascending order)

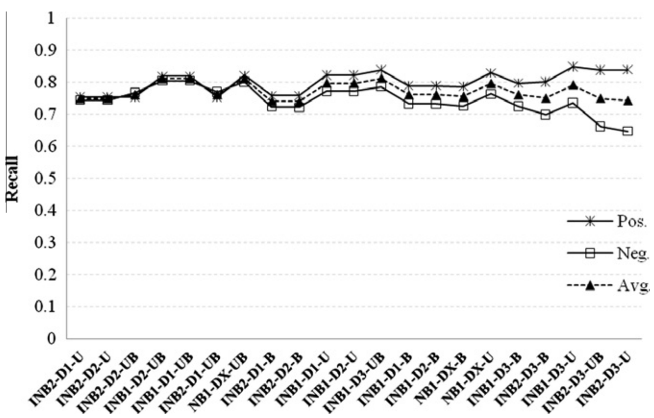

Fig. 4. Recall comparison of the original Naïve Bayes and the improved Naïve Bayes.

[Feature]. For example, NB1 is used as a classifier in NB1-D2-U, and the lexicon used D2, and the Feature used unigrams in the conduct of the experiment. The classifiers that are used in this thesis are NB1, INB-1, INB-2 and S(SVM), and the lexicons are D1, D2, D3 and DX. Here, DX does not apply a lexicon but is tested according to the original method. Additionally, the feature used unigrams (U), bigrams (B), and unigrams + bigrams (UB). Table 9 shows the case that is tested in this thesis. Here, (1) applies the proposed improved Naïve Bayes and the senti-lexicon. Secondly, (2) only applied the senti-lexicon on the original algorithm, SVM. Lastly, (3) used the original algorithm, SVM, and Naïve Bayes, and a lexicon is not applied here.

Fig. 4 shows the recall result in comparison when NB1, INB-1, INB-2 were used as classifiers.

Table 9
A case tested in this thesis.

(1)	(2)	(3)
NB2-D1-U, NB2-D1-B, NB2-D1-UB	S-D2-B, S-D2-UB, S-D3-U	NB1-DX-U, NB1-DX-B
NB2-D2-U, NB2-D2-B, NB2-D2-UB	S-D3-B, S-D3-UB	NB1-DX-UB, S-DX-U
NB2-D3-U, NB2-D3-B, NB2-D3-UB		S-DX-B, S-DX-UB
NB3-D1-U, NB3-D1-B, NB3-D1-UB		S-D1-U, S-D1-B
NB3-D2-U, NB3-D2-B, NB3-D2-UB		S-D1-UB, S-D2-U
NB3-D3-U, NB3-D3-B, NB3-D3-UB		

Here, the difference in the values of the positive classification accuracy and the negative classification accuracy is shown in descending order. It is a INB1-D2-UB and INB1-D1-UB case when the difference in the accuracy values between the two classes (positive or negative) is between 2% and 80%. Two cases show the same accuracy value of 81.3%, and the difference in the accuracy values between the two classes also shows 1.4%. Based on this result, one can see that there is no difference between D1 and D2 in the lexicon, but it is different for INB1-D3-UB when the lexicon D3 is applied. Here, accuracy of 81.4% is shown, but the difference in the accuracy values between the classes is 5.2%. There are cases when the application of the lexicon D3 decreases the accuracy. When the unigrams are used as features, the accuracy of NB1-DX-U that did not use the lexicon was higher than the accuracy of INB1-D3-U and INB2-D3-U, which applied the lexicon D3. In a NB1-DX-UB case that did not apply a lexicon but instead used unigrams + bigrams, the accuracy was 81% but there was a 2.03% difference in the accuracy values of the two classes. When INB-2 was used as a classifier, there was no apparent effectiveness in its accuracy compared to the original method; however, the difference in the accuracy values narrowed to a maximum of 0.9%. Based on the experimental result, the application of a lexicon such as D1 and D2 can be seen as more effective than the original method (NB1-DX-U, NB1-DX-B). Additionally, the accuracy can be increased by using the combination of the unigrams + bigrams (UB) instead of using the unigrams (U) or the bigram (B) individually as features. This showed a synergetic effect in the accuracy even in NB1-DX-UB that did not apply a lexicon in the original Naïve Bayes.

Fig. 5 shows the precision when NB1, INB-1 and INB-2 were used as the classifiers. Here the difference in the values of the positive classification accuracy and the negative classification accuracy are shown in the order of small number, as shown in Fig. 4. In terms of precision, the difference in the accuracy values of the two classes that showed within 3% to above 80% showed the accuracies of 81.3% and 81.2% for INB1-D1-UB and INB1-D2-UB, respectively, and the difference in the accuracy values between the two classes of 2.3% and 2.4% occurred. Here the positive classification accuracy of NB1-DX-B was 97% and the negative classification accuracy was 69.5%, whereby the average accuracy of 83.6% was the highest. However, the difference in accuracy values between two classes was 28.2%, so the result leaned toward the positive. On the other hand, NB1-DX-UB shows an accuracy of 81% and 3.2% difference in the accuracy values between the two classes. In terms of precision, the difference in accuracy values between the two classes was seen as the maximum of 1.98% when INB-2 was used as the classifier.

The experimental result showed that even in terms of precision the accuracy was high when the lexicons D1 and D2 were applied, and the accuracy was high when the unigrams + bigrams was used

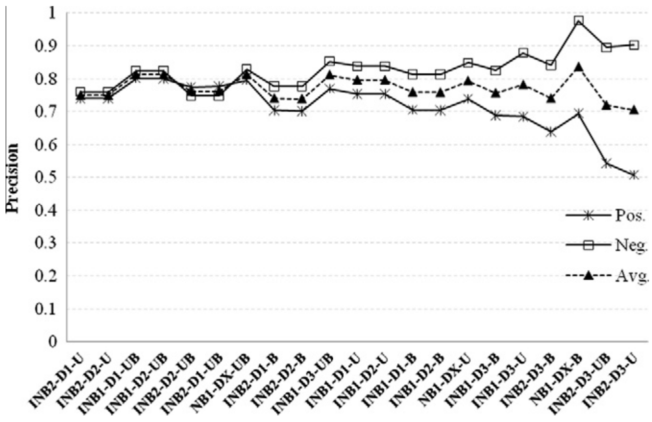

Fig. 5. Precision comparison between the original Naïve Bayes and the improved Naïve Bayes.

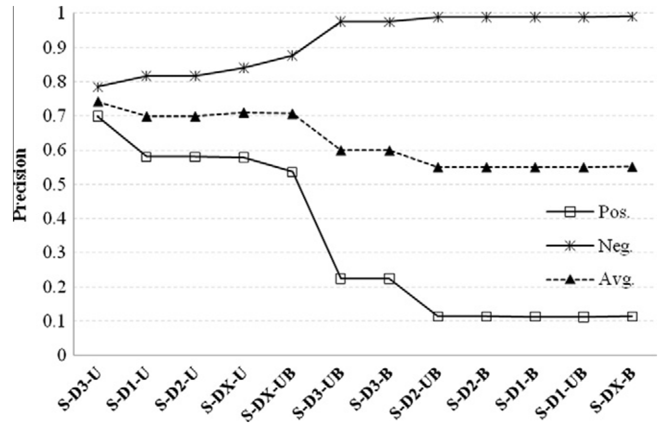

Fig. 7. Precision result of SVM.

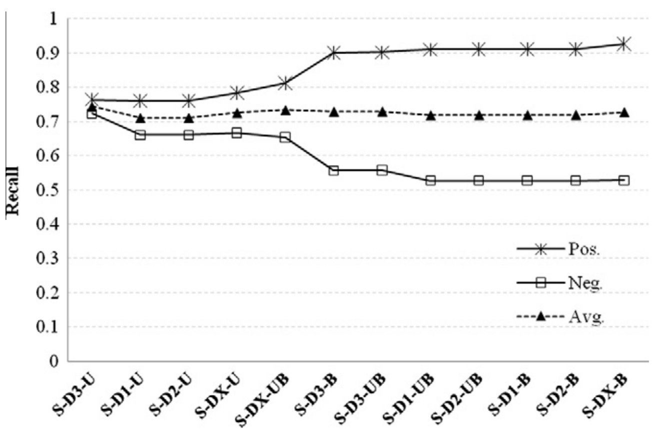

Fig. 6. Recall result of SVM.

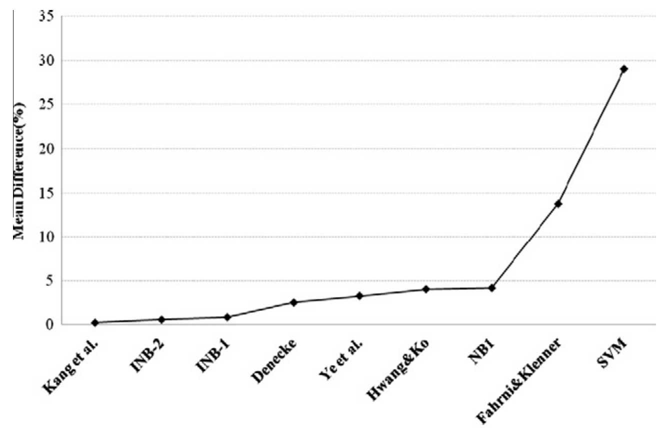

Fig. 8. Comparison of the precision values between the two classes by algorithm.

as the feature. However, as opposed to the recall result, the precision result showed a relatively high accuracy in INB1-D3-UB that applied lexicon D3, and the result was favorable toward the positive in NB1-DX-B.

Fig. 6 shows the recall when SVM was used as the classifier. The SVM algorithm was not more effective than the classification result of NB1, INB-1 and INB-2. In SVM, the unigrams used as the feature in the traditional document classification approach had relatively high accuracy. However, unlike NB1, INB-1 and INB-2, SVM had high classification accuracy when the lexicon D3 was applied.

Fig. 7 shows the precision when SVM is used as a classifier.

When the lexicon D3 is applied and the unigrams is applied here, the accuracy between the two classes is high at a similar level compared to another experimental case.

Fig. 8 shows a comparison against the experimental results of a different research study. The difference in the positive classification accuracy and the negative classification accuracy when each algorithm is used is compared. The selected value here is a calculated average value of the differences in the top three with the smallest gap between the positive classification accuracy and the negative classification accuracy among the many experimental cases of each research result. The reason the top three are selected is because those three had the least number of cases in Ye et al. (2008) and in NB1 of this thesis in each research. In addition, the value of Fig. 8 is an average that is obtained by calculating an F-Measure based on Recall and Precision that are expressed as the experimental result in each research. In an experimental result, Kang et al. (2009) appeared to be the best, with the difference between the accuracies of the two classes of 0.2%. However, if the ones with the average difference within 1% are treated as equivalent, INB-1 is effective in regard to the classification accuracy and the balanced classification. Additionally, INB-2 is better than Kang et al. in the accuracy and the standard deviation of the difference.

Table 10 shows the mean precision, mean of precision difference between the two classes, standard deviation of precision difference shown in Fig. 8.

Table 10
Results of balanced classification.

	Mean precision (%)	Mean difference (%)	Standard deviation of difference
Kang et al.	73.7	0.20	0.26
INB-2	75.4	0.55	0.08
INB-1	81.2	0.83	0.65
Denecke	70.9	2.50	1.60
Ye et al.	77.7	3.25	3.55
Hwang and Ko	70.5	4.01	2.92
NB1	79.7	4.15	4.76
Fahrni and Klenner	48.2	13.77	6.40
SVM	60.0	29.03	20.20

8. Conclusion and future work

In this thesis the issues in the original research were analyzed and the solutions to these problems were sought in order to conduct a sentiment analysis of restaurant reviews. As a solution, the senti-lexicon associated with the restaurant was constructed and an improved Naïve Bayes algorithm was proposed. In order to prove the effectiveness of the proposed algorithm, its performance against the original algorithm was measured through a comparative experiment. In this thesis the original algorithms NB1 and SVM, and the proposed algorithms INB-1 and INB-2 were used to make up the various types of the feature and lexicon, and their performance was compared in each experimental case. As a result, the maximum of 5.6% precision to the maximum of 1.9% performance improvement could be seen in a recall when lexicon D1 and D2 were used and an unigram + bigrams was applied in INB-1. Additionally, the difference between the positive classification accuracy and the negative classification accuracy was narrowed by 3.6% compared to when the original Naïve Bayes were used, and there was a difference of 28.5% compared to when SVM was used. Thus the improved Naïve Bayes algorithm proposed in this thesis proves its effectiveness when the accuracy and balanced classification are considered. Moreover, the unigrams + bigrams feature can be seen as an effective feature for the sentiment analysis of the restaurant review. Thus, as found in the original research that showed various classification results according to the type of feature, lexicon and classification algorithm, this research also showed various results when its subject consisted of restaurant reviews. However, the restaurant reviews showed the best classification effect when INB-1, unigrams + bigrams, and lexicon D1 and D2 were used.

Consequently, it is necessary to analyze the reviews of various disciplines prepared in multiple languages in the future research. In the method proposed in this thesis, features can be extracted with just the POS tagger in each language, and therefore a sentiment analysis of a review in any language is possible.

Acknowledgement

This work was supported by the Korea Research Foundation (KRF) grant funded by the Korea government (MEST) (No. 2010-0015842).

References

Cho, Y., & Lee, K. (2006). Automatic affect recognition using natural language processing techniques and manually built affect lexicon. *IEICE Transactions on Information & Systems*, 2964–2971.

Courses, E., & Surveys, T. (2008). Using SentiWordNet for multilingual sentiment analysis. In *Proceedings of the international conference on data engineering workshop* (pp. 507–512).

Chaumartin, F. (2007). Upar7: A knowledge-based system for headline sentiment tagging. In *Proceedings of the SemEval-2007* (pp. 422–425).

Dave, K., Lawrence, S., & Pennock, D. (2003). Mining the peanut gallery: Opinion extraction and Semantic classification of product reviews. In *Proceedings of the 12th international world wide web conference (WWW '03)* (pp. 512–528).

Denecke, K.(2009). Are SentiWordNet scores suited for multi-domain sentiment classification? In *Proceedings of the 4th international conference on digital information management (ICDIM '09)* (pp. 1–6).

Ding, X., Liu, B., & Yu, P. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the international conference on web search and web data mining (WSDM '08)* (pp. 231–239).

Esuli, A., & Sebastiani, F. (2006). SENTIWORDNET: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th conference on language resources and evaluation* (pp. 417–422).

Fahrni, A., & Klenner, M. (2008). Old wine and warm beer: Target-specific sentiment analysis of adjectives. In *Proceedings of the AISB* (pp. 1-4).

Han, J., & Kamber, M. (2005). *Data mining: Concepts and techniques* (2nd ed.). Morgan Kaufmann Publisher.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD international conference on knowledge discovery and data Mining (KDD '04)* (pp. 168–177).

Hu, Y., & Li, W. (2010). Document sentiment classification by exploring description model of topical terms. *Journal of Computer Speech and Language, 25*, 386–403.

Hwang, J., & Ko, Y. (2008). A korean sentence and document sentiment classification system using sentiment features. *Journal of Korean Institute of Information Scientists and Engineers (KIISE): Computing Practices and Letters, 14*, 336–340.

Kang, H., Yoo, S., & Han, D. (2009). Accessing positive and negative online opinions. In *Proceedings of the 13th international conference on human–computer interaction (HCII2009)* (pp. 359–368).

Kang, H., & Yoo, S. (2007). SVM and collaborative filtering-based prediction of user preference for digital fashion recommendation systems. *IEICE Transactions on Information & System*, 2100–2103.

Kim, S., & Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the international conference on computational linguistics (COLING'04)* (pp. 1–8).

Kim, Y., Jung, Y., & Mayeng, S. (2008). An opinion analysis system using domain-specific lexical knowledge. In *Proceedings of the 4th Asia information retrieval symposium (AIRS 2008)* (pp. 466–471).

Liu, B. (2006). *Web data mining*. Berlin, Heidelberg, New York: Springer. ISBN 978-3-540-37881-5.

Miao, Q., Li, Q., & Dai, R. (2009). AMAZING: A sentiment mining and retrieval system. *Journal of Expert Systems with Applications, 36*, 7192–7198.

Miao, Q., Li, Q., & Dai, R. (2008). An integration strategy for mining product features and opinions. In *Proceeding of the 17th ACM conference on information and knowledge management* (pp. 1369–1370).

Myung, J., Lee, D., & Lee, S. (2008). A Korean product review analysis system using a semi-automatically constructed semantic dictionary. *Journal of KIISE: Software and Applications, 35*, 392–403.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the conference on empirical methods in natural language processing(EMNLP '02)* (pp. 79–86).

Popescu, A.-M., & Etzioni, O. (2005). Extracting product features and opinions from reviews. In *Proceedings of the conference on human language technology and empirical methods in natural language processing (EMNLP'05)* (pp. 339–346).

Prabowo, R., & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informatics, 3*, 143–157.

Sista, S., & Srinivasan, S. (2004). Polarized lexicon for review classification. In *Proceedings of the international conference on machine learning, models, technologies and applications* (pp. 867–872).

Su, Q., Xiang, K., Wang, H., Sun, B., & Yu, S. (2006). Using pointwise mutual information to identify implicit features in customer reviews. In *Proceedings of the international conference on the computer processing of oriental languages* (pp. 22–30).

Tan, S., & Zhang, J. (2008). An empirical study of sentiment analysis chinese documents. *Journal of Expert Systems with Applications, 34*, 2622–2629.

Turney, P.(2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting of the association for computational linguistics* (pp. 417–424).

Turney, P., & Littman, M. (2002). Unsupervised learning of semantic orientation from hundred-billion-word courpus. Technical Report ERB-1094/NRC-44929.

Witten, I., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques* (2nd ed.).

Yang, J., Kim, H., & Lee, S. (2010). Feature-based product review summarization utilizing user score. *Journal of Information Science and Engineering, 26*(1973–1990).

Yang, J., Myung, J., & Lee, S. (2009). A holistic approach to product review summarization, In *Proceedings of 2009 software technologies for future dependable distributed systems* (pp. 150–154).

Ye, Q., Zhang, Z., & Law, R. (2008). Sentiment classification of online reviews to travel destination by supervised machine learning approaches. *Journal of Expert Systems with Applications, 36*, 1–9.

Ye, Q., Shi, W., & Li, Y. (2006). Sentiment classification for movie reviews in Chinese by improved semantic oriented approach. In *Proceedings of the 39th Hawaii international conference on system sciences* (pp. 1–5).

Zhu, J., Xu, C., & Wang, H. (2010). Sentiment classification using the theory of ANNs. *The Journal of China Universities of Posts and Telecommunications, 17*, 58–62.