International Journal of Control and Automation
Vol. 9, No. 9 (2016), pp.331-342
http//dx.doi.org/10.14257/ijca.2016.9.9.32

# Identifying Negative Sentiment with Sentiment Based LDA and Support Vector Machine Classification

Jun-Hui Zheng [1,a] and Gang-Li [2,b *]

*[1]College of Information Engineering, Pingdingshan University, Pingdingshan Henan, China 467000*
*[2]Modern Education Technology Center, Pingdingshan University, Pingdingshan Henan, China 467000*
*[a]pdszhengjunhui@163.com, [b]17179014@qq.com*

## *Abstract*

*With the increasing development of online collaborative platforms, there emerge massive subjective texts. However, due to the massive negative news about eroticism, violence, extremity and corruption, as well as the influences of agitators and provocateurs, it is quite likely that Internet users can be turned from conscious individuals into unconscious groups, which contributes to the accumulation of public negative sentiment. In this work, we focus on the identification of sentiment and especially negative sentiment. Specifically, we introduce sentiment layer to the basic LDA topic model to map the texts into a lower dimensional space of topics and sentiment. Besides, we also consider the sentiment dictionary based sentiment feature word extraction method. By feeding the feature words into Support Vector Machine (SVM) classifier, we get the sentiment tendency of texts. Our experiments prove the efficiency of proposed method.*

***Keywords:*** *Sentiment classification, Topic model, Support Vector Machine (SVM)*

## 1. Introduction

Nowadays, increasing collaborative platforms provide a lot of ways of exchanging information and expressing opinions, such as online communities, microblogs, BBS, and other social supported sites. Accordingly, there emerge massive subjective texts, through which we could better understand the behaviors of users and their opinions of hot events. Sentiment analysis, as one of the most important branches in data mining, deals with and analyzes the opinions, sentiment and attitudes of individuals and groups. It has been successfully applied in many areas, such as marketing analysis, public opinion poll, and information monitoring. For example, sentiment analysis based on the comments of products can aid in making right marketing decisions for enterprises, while sentiment analysis using news comments helps government department better control with the public opinions.

However, due to the massive negative news about eroticism, violence, extremity and corruption, as well as the influences of agitators and provocateurs, it is quite likely that Internet users can be turned from conscious individuals into unconscious groups, which contributes to the accumulation of public negative sentiment. Negative sentiment could affect the emotions, moral values and public opinions of people explicitly or potentially. In order to control and minimize the influences, identifying the negative information and negative sentiment, and therefore guiding the correct information propagation is a significant effort for now.

ISSN: 2005-4297 IJCA

For a long time, Vector Space Model (VSM) [1] is used for text representation, which views documents as a set of topics. However, it has been discovered that VSM takes no consideration of the existence of synonym and the multivocal words, and therefore ignores the semantic relationships between words. Accordingly, Latent Semantic Analysis (LSA) introduces a semantic dimension between documents and words, and extracts the semantic dimension using linear algebra method. Later, probabilistic LSA (pLSA) replaced LSA with probabilistic method. Blei *et al*. [2] proposed Latent Dirichlet Allocation (LDA) model to further improve the topic model. As a mature topic model with thorough mathematics and flexibility, LDA has been used in many text analysis areas.

As one of the most popular classification technique, Support Vector Machine (SVM) was first proposed by Vapnik *et al*. [3]. It is built upon statistics and structural risk minimization principle, and can efficiently solve classification problems with some advantages. For example, SVM can avoid the typical issues in traditional classification algorithms, such as curse of dimensionality, local convergence and over-fitting. Therefore, in this paper, we employ SVM as the classifier.

Specifically, in this paper, we propose to integrate the sentiment word dictionary based method with LDA based method to better extract the negative sentiment of texts. On one hand, sentiment words are extracted based on existing sentiment dictionaries; on the other hand, to capture the latent semantic information of texts, we modify the LDA model by introducing sentiment information, and then we learn the topic based representation with sentiment embedded words for texts. Combining above two kinds of information together, we get an extended set of feature sentiment words. Feed the feature words into SVM classification model, we get the integrated global sentiment of texts.

## 2. Related Work

Existing efforts on sentiment classification in text mining can be categorized into two groups. The first group of methods is the typical feature selection method. For example, Chidanand *et al*. [4] selected feature words based on word frequency, and Yang [5] used chi square statistics and information gain. The second category is to use polarity words. For example, Fu *et al*. [6] proposed a sentiment analysis model based on fuzzy set theory by considering multiple granularity of sentiment. Guo *et al*. [7] considered high frequency words and basic sentiment features. Zhang [8] employed HowNet to label sentiment words and calculate the similarity between words.

Another effort is to employ topic models for sentiment analysis. It is indicated that sentiment is dependent on topic information [9]. Jo *et al*. [10] proposed the Aspect and Sentiment Unification Model (ASUM), and Lin *et al*. [11] designed the Joint Sentiment/Topic (JST) model. However, these two models view topic or sentiment as single description of texts. Men *et al*. [12] proposed a Topic Sentiment Mixture (TSM) by simulating the mixture of topic and sentiment. However, it doesn’t consider the sentiment distribution for documents. To this end, in this paper, we introduce a sentiment layer into LDA model to completely capture the sentiment information for texts.

## 3. Preliminaries

### 3.1. LDA Model

LDA model, first proposed by Blei [2], is a probability model built on a discrete dataset such as a set of documents. LDA model is clearly structured and widely applied on text classification, text modeling and information retrieval, *etc*. Specifically, using a three-layer

Bayes generative model of document-topic-word, LDA maps the high dimensional set of documents into lower dimensional latent topic space. That is, LDA views documents as the mixture of topics, and topics as the distribution over word space. Figure 1 illustrates the graphical representation of LDA, and the notations are listed in Table 1.

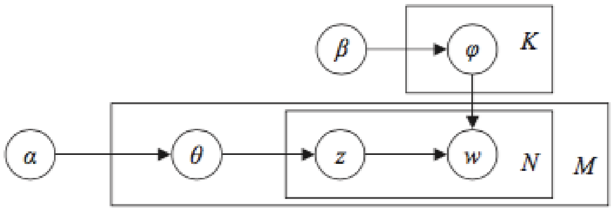

**Figure 1. LDA Model**

**Table1. Notation of LDA**

| Notation | Description |
|---|---|
| $N$ | Number of words in a document |
| $M$ | Number of documents |
| $K$ | Number of topics |
| $j$ | Probability distribution of topic-word |
| $q$ | Probability distribution of document-topic |
| $z$ | Probability distribution of topics |
| $w$ | Word |
| $a$ | Hyper parameter of $q$ |
| $b$ | Hyper parameter of $j$ |

Suppose $q_d, j_z$ follows Dirichlet distribution with parameters $a, b$ respectively, where $q_d$ is the probability distribution of topics on document $d$, and $j_z$ is the probability distribution of words on topic $z$. The probability of $i$-th word in document $d$ is calculated as:

$$P(w_i) = \overset{K}{\underset{j=1}{\mathring{a}}} P(w_i \mid z_i = j)P(z_i = j), \tag{1}$$

where $z_i$ is the latent variable, $z_i = j$ is the feature word selected from topic $j$, $P(w_i \mid z_i = j)$ is the probability of word $w_i$ belonging to topic $j$, and $P(z_i = j)$ is the probability of topic $j$ belonging to document $d$.

Suppose $j_w^{z=j} = P(w_i \mid z_i = j)$ is the multinomial distribution of topic $j$ over words, and $q_{z=j}^d = P(z_i = j)$ is the multinomial distribution of document $d$ over topics. Then, the probability of word $w_i$ in $d$ can be calculated as:

$$P(w|d) = \overset{K}{\underset{j=1}{\mathring{a}}} j_w^{z=j} q_{z=j}^d . \tag{2}$$

LDA has some prior probability hypothesis on $j^z, q^d$, which follow the Dirichlet distribution. The probability models are:

$$w_i \,|\, z_i, q^{z_i} \square\, Multinomial(q^{z_i}), \tag{3}$$

$$z_i \,|\, q^{d_i} \square\, Multinomial(q^{d_i}), \tag{4}$$

$$j^{\,z_i} \square\, Dirichlet(b), \tag{5}$$

$$q^{d_i} \square\, Dirichlet(a), \tag{6}$$

where $a, b$ are hyper parameters of Dirichlet distribution.

Each document $d$ is generated as follows:

Step 1: select $N \square\, Possion(x)$, where $N$ is the size of $d$;

Step 2: select $q \square\, Dirichlet(a)$, where $q$ is the document-topic distribution;

Step 3: select $j \square\, Dirichlet(b)$, where $j$ is the topic-word distribution;

Step 4: for each word in $N$: (1) generate topic $z \square\, Multinomial(q)$; (2) generate word by $p(w \,|\, z, j)$.

The inference of LDA is to solve $q, z$ given document $d$, that is,

$$P(q, z \,|\, w, a, b) = \frac{P(q, z, w \,|\, a, b)}{P(w \,|\, a, b)}. \tag{7}$$

One of the most popular inference methods is to use Gibbs Markov-chain Monte Carlo (MCMC) sampling [13]. Specifically, the sampling process is:

Step 1: initialization: fro each word $w_i$, assign a topic randomly;

Step 2: update: for each word $w_i$, given the topics of other words other than $w_i$, notated as $z_{-i}$, calculate the poster probability of $w_i$ belonging to topic $j$, that is, $P(z_i = j \,|\, z_{-i}, w)$, in order to assign $w_i$ to the most likely topic;

Step 3: iteratively perform Step 2 until converges:

$$P(z_i \,|\, z_{-i}, w) = \frac{n_{z_i}^{w} - 1 + b}{n_{z_i} - 1 + Nb} \times \frac{n_d^{z_i} - 1 + a}{n_d - 1 + Ka}, \tag{8}$$

where $N, K$ are the numbers of words and topics, $n_{z_i}^{w}$ is the number of $w$ being assigned to $z_i$, $n_{z_i}$ is the total number of all words assigned to $z_i$, $n_d^{z_i}$ is the number of words in $d$ being assigned to $z_i$, and $n_d$ is the total number of words in $d$. The former part is the ratio of word $w_i$ belonging to topic $j$, and the latter part is the ratio of words in document $d$ assigned to topic $j$.

After many sampling iterations, we have the topic assignment of each word and each topic appearance of all documents:

$$j_{z,w} = \frac{n_z^{w} + b}{n_z + Nb}, \tag{9}$$

$$q_{d,z} = \frac{n_d^{z} + a}{n_d + Ka}. \tag{10}$$

### 3.2. SVM Basics

The basic idea of SVM is to construct a classifier such that the margin between different groups is maximized and the error rate of classification is minimized. Figure 2 illustrates the binary classification of SVM, where the hyperplane between $H_1, H_2$ separates the data with the maximum margin.

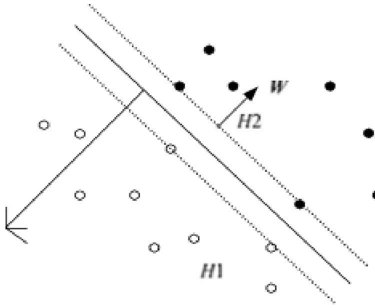

**Figure 2. Illustration of SVM Classification**

Suppose the sample dataset $\{(x_1,y_1),(x_2,y_2),...,(x_n,y_n)\}$, where $x_i\ \text{Î}\ R^n, y_i\ \text{Î}\ \{+1,-1\}$. The hyperplane $w^T x + b = 0$ separates the data point into two groups, where $w$ is the vertical direction of the hyperplane, and $b/|w|$ is the distance between the origin and the hyperplane. Let $l^+, l^-$ be the nearest distance between the hyperplane and the positive and negative data points. SVM can find the hyperplane which maximizes $l^+$ and $l^-$. The model can be formulated as:

$$\begin{cases} w^T x_i + b \ge 1, & y_i = 1; \\ w^T x_i + b \le -1, & y_i = -1. \end{cases} \qquad (11)$$

That is,

$$y_i(w^T x_i + b)\ {}^3\ 1, l^+ = l^- = \frac{1}{||w||}. \qquad (12)$$

The problem can be transformed as a quadratic programming problem:

$$\begin{aligned} &\min\ ||w||^2 \\ &\text{s.t.}\ \ y_i(w^T x_i + b)\ {}^3\ 1 \end{aligned} . \qquad (13)$$

Accordingly, the dual problem is:

$$\begin{aligned} &\max\ Q(a) = \mathop{\text{å}}\limits_{i=1}^{n} a_i - \frac{1}{2}\mathop{\text{å}}\limits_{i=1}^{n}\mathop{\text{å}}\limits_{j=1}^{n} a_i a_j y_i y_j x_i^T x_j \\ &\text{s.t.}\ \ \mathop{\text{å}}\limits_{i=1}^{n} a_i y_i = 0, a_i\ {}^3\ 0 \end{aligned} . \qquad (14)$$

Solve above problem, we get:

$$w = \mathop{\text{å}}\limits_{i=1}^{n} a_i y_i x_i, b = y_i - \mathop{\text{å}}\limits_{i=1}^{n} y_i a_i x_i^T x_j . \qquad (15)$$

## 4. Proposed Method

In this section, we present our proposed method for classifying negative sentiment for texts. The basic idea is to extract a list of feature words and feed them into a classifier to learn the label of sentiment orientation for texts. Specifically, introduce sentiment layer into LDA model to get the sentiment based topic distribution; besides, we also consider the sentiment word dictionary to extend the sentiment feature words learned from sentiment based topic model. Then, SVM model is employed for classification based on above feature words to get the global sentiment of texts.

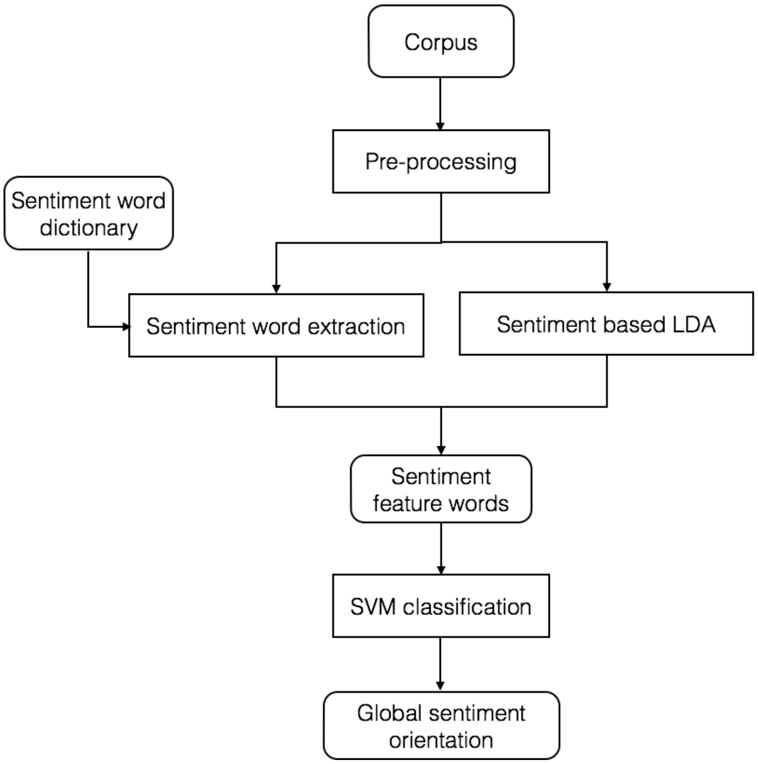

**Figure 3. Workflow of Identifying Negative Sentiment**

The overall workflow is shown in Figure 3. After preprocessing the original corpus, two steps are conducted in parallel. On one hand, sentiment words are extracted based on existing sentiment dictionaries; on the other hand, to capture the latent semantic information of texts, we modify the LDA model by introducing sentiment information, and then we learn the topic based representation with sentiment embedded words for texts. After that, combining above two kinds of information together, we get an extended set of feature sentiment words. Feed the feature words into SVM classification model, we get the integrated global sentiment of texts.

### 4.1. Sentiment Word Extraction

In this step we use sentiment word dictionary to extract sentiment tendency for sentences. The basic idea is to calculate the similarity between words in dictionary to determine the sentiment tendency of sentences. First, we construct a sentiment dictionary library with baseline words, negative words, and degree modifiers. Then, perform words segmentation and Part-Of-Speech (POS) tagging. In this work, we employ the Chinese segmentation system ICTCLAS [14] for words segmentation and POS tagging. Lastly, considering negative words and degree modifiers, determine the sentiment tendency of sentences.

Suppose the baseline positive and negative words are $base_p$ and $base_n$ respectively. Given word $w$, the sentiment tendency can be calculated as:

$$st(w) = \sum_{i=1}^{k} sim(base_{pi}, w) - \sum_{i=1}^{k} sim(base_{ni}, w), \tag{16}$$

where $k$ is the number of baseline word pairs in the dictionary, and $sim(\cdot)$ is the similarity function.

We consider the negative words typically in front of other words, such as no, never, seldom, not, hardly, *etc.*, which can completely reverse the sentiment tendency of sentences. If there exist negative words in front of sentiment words, the sentiment tendency is updated by:

$$st(w_{-n}) = (-1)^a \frac{1}{2}\sqrt{st(w)}, \tag{17}$$

where $a$ is the appearance of negative words in $w$, and $w_{-n}$ is $w$ without negative words.

Another case is degree modifiers, which define different degrees of sentiment. For example, “extremely large” is much stronger than “large”. If there exist degree modifier words, the sentiment tendency is updated by:

$$st(w_d) = st(w) \cdot \deg(d), \tag{18}$$

where $d$ is the degree modifier in $w$, $\deg(d)$ is the strength of $d$, which is manually defined in this work, and $w_d$ is the word modified by $d$.

The final sentiment tendency of document $d$ is the combination of Equations (16), (17) and (18):

$$st(d) = \sum st(w) + \sum st(w_{-n}) + \sum st(w_d), \tag{19}$$

where $w$ is the word in document $d$.

The sentiment tendency of $d$ based on sentiment dictionary and sentiment word extraction, *i.e.*, $st(d)$, would be used as one of the features for the input vector of SVM classifier.

### 4.2. Sentiment based LDA Model

In this step we construct a sentiment based LDA model to learn the sentiment embedded topic distribution of sentences. The basic assumption is that not only there exists a latent topic layer between documents and words, but also a sentiment layer.

Therefore, we introduce a sentiment layer into the LDA model, as shown in Figure 4, where $s$ denotes sentiment, $\pi$ is the probability distribution of document-sentiment, and $\gamma$ is the hyper parameter of $\pi$. Note that in this work we only consider sentiment as positive or negative, and therefore $\pi$ follow Bernoulli distribution.

Each document $d$ is generated as follows:

Step 1: select $N \sim Possion(\xi)$, where $N$ is the size of $d$;

Step 2: select $\theta \sim Dirichlet(\alpha)$, where $\theta$ is the topic distribution;

Step 3: select $j$ □ $Dirichlet(b)$, where $j$ is the topic-word distribution;

Step 4: for each word in $N$: (1) select topic $z$ □ $Multinomial(q)$; (2) select sentiment $s$ □ $Bernoulli(p)$; (3) select word by $P(w\,|\,z,s,j)$.

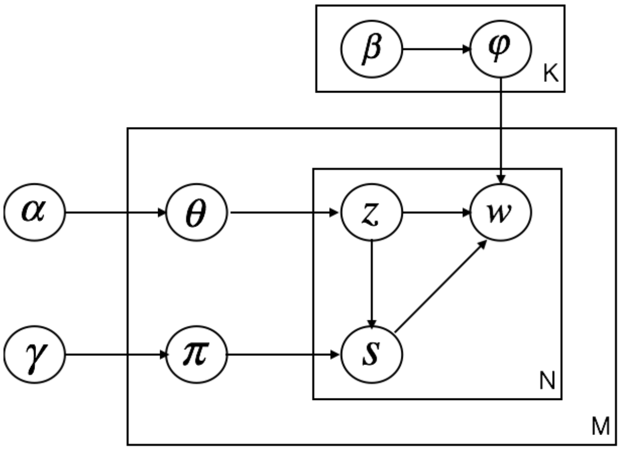

**Figure 4. Graphical Representation of Sentiment Based LDA**

The joint probability of word, topic and sentiment is:

$$P(w,s,z) = P(w\,|\,s,z)P(s,z) = P(w\,|\,s,z)P(s\,|\,z)P(z), \tag{20}$$

where

$$P(w\,|\,s,z) = \left(\frac{G(Vb)}{\prod_{v=1}^{V} G(b)}\right) \frac{G\left(\sum_{w=1}^{W}(C_{z=k,s=j} + b_{s=j,w})\right)}{G\left(\sum_{w=1}^{W}(C_{z=k,s=j} + b_{s=j,w}) + C_d\right)} \prod_{w=1}^{W} \frac{G(C_{z=k,s=j} + b_{s=j,w} + C_{\|w_d\|})}{G(C_{z=k,s=j} + b_{s=j,w})}, \tag{21}$$

where $C_{z=k,s=j}$ is the number of words with topic $k$ and sentiment $j$, $C_d$ is the number of words in $d$, $C_{\|w_d\|}$ is the number of $w$ in $d$, $b_{s=j,w}$ is the prior distribution of $w$, and

$$P(s\,|\,z) = \left(\frac{G(\sum_{j=1}^{S} g_{z=k,j})}{\prod_{j=1}^{S} G(g_{z=k,j})}\right)^{D\times K} \prod_{d=1}^{D}\prod_{k=1}^{K} \frac{\prod_{j=1}^{S} G\left(C_{d,z=k,s=j} + g_{z=k,j}\right)}{G(C_{d,z=k} + \sum_{j=1}^{S} g_{z=k,j})}, \tag{22}$$

where $C_{d,z=k}$ is the number of documents with topic $k$, $C_{d,z=k,s=j}$ is the number of documents with topic $k$ and sentiment $j$, $g_{z=k,j}$ is the prior distribution of sentiment $j$, and

$$P(z)=\left(\frac{\Gamma(\sum_{k=1}^{K}\alpha_k)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\right)^{D}\prod_{d=1}^{D}\frac{\prod_{k=1}^{K}\Gamma(C_{d,z=k}+\alpha_k)}{\Gamma(C_d+\sum_{k=1}^{K}\alpha_k)}, \tag{23}$$

where $C_d$ is the number of documents, and $C_{d,z=k}$ is the number of documents with topic $k$.

After inference and calculation, we estimate the parameters as follows:

$$\hat{\theta}_{d,j,k}=\frac{C_{d,j}^{k}+\alpha_k}{\sum_{k=1}^{K}(C_{d,j}^{k}+\alpha_k)}, \tag{24}$$

$$\hat{\varphi}_{k,j,w}=\frac{C_{k,j}^{w}+\beta_w}{\sum_{w=1}^{W}(C_{k,j}^{w}+\beta_w)}, \tag{25}$$

$$\hat{\pi}_{d,j}=\frac{C_{d}^{j}+\gamma_j}{\sum_{j=1}^{S}(C_{d}^{j}+\gamma_j)}, \tag{26}$$

where $C_{d,j}^{k}$ is the number of words of document $d$ with topic $k$ and sentiment $j$, $C_{k,j}^{w}$ is the number of assignment of $w$ belonging to topic $k$ and sentiment $j$, and $C_{d}^{j}$ is the number of document $d$ assigned to sentiment $j$.

After learning the topic and sentiment distribution of document, we construct a feature vector prepared for the input of SVM.

## 5. Experiment

We crawl data from Sina and Sohu news with three categories: sports, education and IT. We collect 10,498 news totally along with their comments, among which 2,045 in sports, 3,896 in education and 4,557 in IT. The ground-truth is manually labeled.

We employ precision, recall and F-measure to evaluate the performance of classification:

$$P=\frac{TP}{TP+FP}, \tag{27}$$

$$R=\frac{TP}{TP+TN}, \tag{28}$$

$$F=\frac{2\times P\times R}{P+R}, \tag{29}$$

where $TP,FP,TN$ denotes the number of true positive, false positive and true negative in the classification results.

Table 2 compares the results of (1) SVM using VSM model (VSM+SVM), (2) SVM using basic LDA (LDA+SVM), and (3) SVM using our proposed method (OUR+SVM).

We can see that our method outperforms other two methods in all three measures, while the VSM solution is the worst.

**Table 2. Comparison of Sentiment Classification Results**

| Dataset | VSM+SVM | | | LDA+SVM | | | OUR+SVM | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| Sports | 0.7612 | 0.7058 | 0.7325 | 0.8025 | 0.8110 | 0.8067 | 0.8394 | 0.8265 | 0.8329 |
| Education | 0.8134 | 0.8026 | 0.8080 | 0.8445 | 0.8379 | 0.8412 | 0.8517 | 0.8491 | 0.8504 |
| IT | 0.8578 | 0.8481 | 0.8529 | 0.8632 | 0.8590 | 0.8611 | 0.8799 | 0.8703 | 0.8751 |

## 6. Conclusion

In this work, we research on the sentiment tendency classification problem in text mining area, and especially the negative sentiment. Specifically, we propose to introduce sentiment layer into the typical LDA model. Besides, we integrate topic model and the dictionary based feature words extraction method together, to construct the feature vector for given sentence or document, which would be later fed into the SVM classifier. Our experiment shows that our method can achieve good performance.

In future works, we would like to explore real time negative sentiment detection to extend this work, which is indeed significant for public opinion management and crisis response.

## References

[1] G. Salton, A. Wong and C. S. Yang, “A Vector Space Model for Automatic Indexing”, Communications of the Acm Cacm Homepage, vol. 18, no. 10, **(1974)**, pp. 613-620.

[2] M. I. Jordan, D. M. Blei and A. Y. Ng, “Latent Dirichlet Allocation”, Journal of Machine Learning Research, vol. 3, **(2003)**, pp. 465-473.

[3] V. N. Vapnik and D. Wu, “Support Vector Machine for Text Categorization”, AT&T Research Labs, http://citeseer.nj.nec.com/347263.htm, no. 1, pp. 103c, **(1998)**.

[4] Apté, Chidanand, F. Damerau, and S.M. Weiss, “Automated learning of decision rules for text categorization”, ACM Transactions on Information Systems (TOIS), vol. 12, no. 3, **(1994)**, pp. 233-251.

[5] Y. Yang and J. O. Pedersen, “A Comparative Study on Feature Selection in Text Categorization”, Proc Icml, vol. 4, **(1997)**, pp. 412--420.

[6] G. Fu and X. Wang, “Chinese sentence-level sentiment classification based on fuzzy sets”, Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Association for Computational Linguistics, **(2010)**.

[7] Guo and Honglei, “Domain customization for aspect-oriented opinion analysis with multi-level latent sentiment clues”, Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, **(2011)**.

[8] Zhang and Changli, “Sentiment analysis of Chinese documents: From sentence to document level”, Journal of the American Society for Information Science and Technology, vol. 60, no. 12, **(2009)**, pp. 2474-2487.

[9] C. Lin, Y. He and R. Everson, “A comparative study of Bayesian models for unsupervised sentiment detection”, Proceedings of, **(2010)**, pp. 144--152.

[10] Y. Jo and A. H. Oh, “Aspect and sentiment unification model for online review analysis”, Proceedings of the fourth ACM international conference on Web search and data mining. ACM, **(2011)**, pp. 815-824.

[11] C. Lin and Y. He, “Joint sentiment/topic model for sentiment analysis”, In Proceedings of the 18th ACM international conference on information and knowledge management (CIKM, **(2009)**, pp. 375-384.

[12] Q. Mei X. Ling and M. Wondra, “Topic sentiment mixture: modeling facets and opinions in weblogs”, Proc.of Int.conference on World Wide Web, **(2007)**, pp. 171--180.

[13] P. M. Djuric and J. H. Chun, “An MCMC sampling approach to estimation of nonstationary hidden Markov models”, Signal Processing IEEE Transactions on, vol. 50, no. 5, **(2002)**, 1113--1123.

[14] H. P. Zhang, H. K. Yu and D. Y. Xiong, "HHMM-based Chinese lexical analyzer ICTCLAS", Proc of Sighan Workshop on Chinese Language Processing, vol. 17, **(2003)**, pp. 184-187.

## Authors

**Jun-Hui Zheng**, he received his M.Sc. in Information Sciences (2010) from University. Now he is lecturer of informatics at College of computer science and technology, PDS University. His current research interests include Artificial Intelligence and Control Techniques.

**Gang-Li**, he received his M.Sc. in Information Sciences (2008) from University. Now he is lecturer of Modern Educational Technology Center, PDS University. His current research interests include Artificial Intelligence and Computer network security.