Sentiment Classification with Support Vector Machines and Multiple Kernel Functions

Tanasanee Phienthrakul[1], Boonserm Kijsirikul[2], Hiroya Takamura[3], and Manabu Okumura[3]

[1] Department of Computer Engineering, Faculty of Engineering, Mahidol University, 25/25 Phutthamonthon, Salaya, NakornPathom, 73170 Thailand
[2] Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, 254 Phyathai Road, Patumwan, Bangkok 10330 Thailand
[3] Precision and Intelligence Laboratory, Advanced Information Processing Division, Tokyo Institute of Technology, 2459 Nagatsuta, Midori-ku, Yokohama, 226-8503 Japan
tanasanee@yahoo.com, boonserm,k@chula.ac.th, takamura@pi.titech.ac.jp, oku@pi.titech.ac.jp

Abstract. Support vector machine (SVM) is a learning technique that performs well on sentiment classification. The performance of SVM depends on the used kernel function. Hence, if the suitable kernel is chosen, the efficiency of classification should be improved. There are many approaches to define a new kernel function. Non-negative linear combination of multiple kernels is an alternative, and the performance of sentiment classification can be enhanced when the suitable kernels are combined. In this paper, we analyze and compare various non-negative linear combination kernels. These kernels are applied on product reviews to determine whether a review is positive or negative. The results show that the performance of the combination kernels that outperforms the single kernels.

Keywords: Sentiment Classification, Support Vector Machine, Evolutionary Strategy.

1 Introduction

Sentiment classification has been a focus of recent research. It has been applied on different domains such as movie reviews, product reviews, and customer feedback reviews [1]. The most basic task in sentiment classification is to classify a document into positive or negative sentiment. The difficulty of sentiment classification is the context-dependency of the sentiments of linguistic expressions. For example, negation words such as “not” or “never” shift the sentiment. A positive statement becomes negative when it is subcategorized by a verb “doubt”. Although we could use n-grams (continuous n words) as features in order to handle such shifts, dependency between two words with a long distance cannot be captured by n-grams. Instead of using n-grams, we utilize higher-degree kernel functions, which can automatically take into account the conjunctions of features.

Support vector machine (SVM) is a kernel method that is successful in many different fields. There are many pieces of research work that applied SVM on sentiment

C.S. Leung, M. Lee, and J.H. Chan (Eds.): ICONIP 2009, Part II, LNCS 5864, pp. 583–592, 2009.

classification problems. The results show that SVM yields the best result when it is compared with other approaches such as in [2] and [3]. SVM classifier can be trained using a large number of features [2]. Besides, the learning process of SVM optimizes the margin between two classes on feature space by using the kernel techniques. We however confront another difficulty that it is unknown which kernel function is suitable for this task. In spite of our intuition that feature combinations will capture the context-dependency of sentiment, some researchers have reported that higher-degree kernels only degraded the classification performance [4], [5]. Meanwhile, there is a report that higher-degree kernels did improve the classification performance [6]. This lead us to the idea of using the non-negative linear combination of multiple kernels with adjusted weight parameters.

In using the combination, we need to determine the weight on each kernel in the combination. In order to obtain the suitable weights, we propose to use an evolutionary strategy. The objective function is an important part in evolutionary algorithms, and there are many ways to measure the fitness of the parameters. Training accuracy or training error is the basic function that can be used for evaluating the parameters. Although this function is very easy to calculate, it may cause the overfit to training data. Hence, we propose to estimate the generalization performance of the learning model by the bound that can be derived from the stability property. It is a tight bound, and can be a good criterion for evaluating the parameters in the evolutionary process.

In this paper, we propose the non-negative linear combination kernels and their normalization in Section 2. There parameters are adjusted by the evolutionary strategies that is illustrated in Section 3. Our approach is tested on sentiment classification and the results are reported in Section 4, and the last section is the conclusion.

2 Non-negative Linear Combination Kernels

The support vector machine is a classifier which finds the optimal separating hyperplane in terms of some generalization criterion [7]. In the simple pattern recognitions, SVM uses a linear separating hyperplane to create a classifier with the maximum margin [8]. In soft margin SVM, the width of margin can be controlled by a regularization parameter C [8]. The constant $C > 0$ determines the trade-off between margin maximization and training error minimization [9]. For non-linear problems, there is an important technique, called *kernel method*, which enables these machines to produce complex nonlinear boundaries inside the original space. This is performed by mapping the input space into a higher dimensional feature space through a mapping function Φ and separating there [9]. However, in SVM, it is not necessary to know the explicit form of Φ . Only the inner product in the feature space, $K(x, y) = \Phi(x) \cdot \Phi(y)$ called *kernel function*, must be defined.

There are many functions that can be used as the kernel in SVM. These kernels are suitable for some problems, and they must be chosen for the tasks under consideration by hand or using prior knowledge [8]. In this paper, we take an interest in four kernel functions i.e. linear, polynomial, RBF, and sigmoid kernels. These kernel functions are shown in Table 1. In order to obtain a better result, the non-negative linear combination of kernels is proposed for SVM on sentiment classification. The analytic expression of this kernel is the following:

$$\mathrm{K}(x, y) = \sum_{i=1}^{n} a_i K_i(x, y), \tag{1}$$

where n is the number of sub-kernels, $a_i \geq 0$ for $i = 1,...,n$ are the arbitrary non-negative weighting constants, and $K_i(x, y)$ for $i = 1,...,n$ are the sub-kernels.

Table 1. Common Kernel Functions

Kernel	Formula
Linear	$K(x, y) = x \cdot y$
Polynomial	$K(x, y) = (x \cdot y + c)^d$
Gaussian RBF	$K(x, y) = \exp(-\gamma \|x - y\|^2)$
Sigmoid*	$K(x, y) = \tanh(x \cdot y + c)$

*This kernel may not be Mercer's kernel.

The linear, polynomial, and RBF kernels are the well-known Mercer's kernels. Therefore, the non-negative linear combination of these kernels still corresponds to the Mercer's theorem. Although it is known that the sigmoid function may not be Mercer's kernel, the sigmoid function is quite popular for support vector machines. Therefore, the kernel functions that will be combined to create a new kernel function are chosen from these four popular kernels. The examples of non-negative linear combination kernels are shown in (2), (3), and (4). These kernels are more flexible as it has more adjustable parameters.

$$\mathrm{K}(x, y) = a_1 K_{Linear}(x, y) + a_2 K_{Poly}(x, y) + a_3 K_{RBF}(x, y) + a_4 K_{Sigmoid}(x, y) \tag{2}$$

$$\mathrm{K}(x, y) = a_1 K_{Linear}(x, y) + a_2 K_{Poly}(x, y) + a_3 K_{RBF}(x, y) \tag{3}$$

$$\mathrm{K}(x, y) = a_1 K_{Linear}(x, y) + a_2 K_{Poly}(x, y) \tag{4}$$

Besides, the non-negative linear combinations of polynomial and RBF kernels are proposed. With these two kind of kernel functions, there are three possible ways of non-negative linear combinations in order to combined them, i.e., (1) the non-negative linear combination of several polynomial kernels at different degree, (2) the non-negative linear combination of several RBF kernels at different scale, and (3) the non-negative linear combination of both polynomial and RBF kernels at different parameters.

Then, normalization is an important pre-processing [10]. Normalization in feature space is not applied directly on the input vector, but it can be seen as a kernel interpretation of the preprocessing [11]. This normalization redefines a new kernel function $\tilde{\mathrm{K}}(x, y)$ of SVM. The non-negative linear combination kernels are normalized by

$$\tilde{\mathrm{K}}(x, y) = \frac{\mathrm{K}(x, y)}{\sqrt{\mathrm{K}(x, x)\,\mathrm{K}(y, y)}}. \tag{5}$$

The normalized kernels place the data on a portion of the unit hypersphere in the feature space [10]. Obviously, the equation $\tilde{\mathrm{K}}(x, x) = 1$ holds true.

3 Weight Adjustment

In the non-negative linear combination kernels, the weight of each sub-kernel is the adjustable parameters. These parameters can be determined by executing a grid search and measuring some goodness criterion at each point (criteria are discussed in Section 3.4), but this kind of search consumes a lot of time especially when there are multiple adjustable parameters. The evolutionary strategy (ES, [12]) is a method that can efficiently find the optimal parameters; it is based on the principles of adaptive selection found in the natural world. Each generation (iteration) of the ES algorithm takes a population of individuals (potential solutions) and modifies the problem parameters to produce offspring (new solutions) [13]. Only the highest fit individuals (better solutions) survive to produce new generations [13]. This algorithm has been successfully used to solve various types of optimization problems [14]. Hence, we propose to use the evolutionary strategies for choosing the parameters of non-negative linear combination kernels and SVM.

There are several different versions of ES. Nevertheless, we prefer to use the (5+10)-ES where 5 solutions are used to produce 10 new solutions by a recombination method. These new solutions are mutated and evaluated, and only the 5 fittest solutions are selected from 5+10 solutions to be the parents in the next generation. These processes will be repeated until a fixed number of generations have been produced or the acceptance criterion is reached.

3.1 Initialization

Let $\vec{v}$ be the non-negative real-value vector of the parameters. The vector $\vec{v}$ has $n+1$ dimensions and it is represented in the form:

$$\vec{v} = (C,\ a_1,\ a_2,\ \ldots,\ a_n), \tag{6}$$

where C is the regularization parameter, a_i for $i = 1, \ldots, n$ are the weights of each sub-kernel, and n is the number of terms of sub-kernel.

For multiple RBF sub-kernels, the width of RBF will be added into the vector of parameters ($\vec{v}$). They also will be investigated by the evolutionary strategy. The (5+10)-ES algorithm starts with the 0^{th} generation (t=0) that selects 5 solutions ($\vec{v}_1, \ldots, \vec{v}_5$) and standard deviation $\vec{\sigma} \in R_+^{n+1}$ using randomization. These 5 initial solutions are evaluated to calculate their fitness. Our goal is to find $\vec{v}$ that optimizes the objective function $f(\vec{v})$ that will be carefully designed in Section 3.4.

3.2 Recombination

For each generation, the 5 fittest solutions are assigned the probabilities of selection to create new solutions. These fittest solutions are ordered by their objective functions, i.e. $\vec{v}_i$ is fitter than $\vec{v}_{i+1}$. Then, their probabilities are assigned by

$$\text{Prob}(\vec{v}_i) = \frac{\mu-(i-1)}{\sum_{j=1}^{\mu} j} = \frac{\mu-(i-1)}{\mu(\mu+1)/2} = \frac{2}{\mu}\left(1-\frac{i}{\mu+1}\right) \tag{7}$$

for $i = 1, 2, \ldots, \mu$, when μ is the number of fittest solutions. In this case, μ is equal to 5.

After that, any 2 solutions are randomly selected from the conventional 5 solutions with their probabilities. Then, the average of this pair of solutions, element by element, is a new solution. This method is called the global intermediary recombination method, and it will be used to create 10 new solutions.

3.3 Mutation

The $\vec{v}_i'$ for $i = 1,\ldots,10$ are mutated by adding each of them with $\vec{z}$ where

$$\vec{z} = (z_1, z_2, \ldots, z_{n+1}) \tag{8}$$

when z_i is a random value from a normal distribution with zero mean and σ_i^2 variation.

$$\begin{gathered} mutate(\vec{v}) = (C + z_1, a_1 + z_2, a_2 + z_3, \ldots, a_n + z_{n+1}) \\ z_i \sim N_i(0, \sigma_i^2) \end{gathered} \tag{9}$$

Moreover, in each generation, the standard deviation will be adjusted by (14) when τ is an arbitrary constant.

$$\begin{gathered} mutate_{\sigma}(\vec{\sigma}) = (\sigma_1 \cdot e^{z_1}, \sigma_2 \cdot e^{z_2}, \ldots, \sigma_{n+1} \cdot e^{z_{n+1}}) \\ z_i \sim N_i(0, \tau^2) \end{gathered} \tag{10}$$

3.4 Objective Function

In general, training error can be used as the objective function in the evolutionary processes. However, this function may cause the overfit to training data. Sometimes, data contain a lot of noise, and thus if the model fits these noisy data, the learned concept may be wrong. Hence, this paper proposed to use the bound of generalization error that is derived from the assumption of stability. The concept of stability was proposed by Bousquet and Elisseeff [15]. They defined the notions of stability for learning algorithms and showed how to use the notions to derive generalization error bounds [15]. In this work, the stability of soft margin SVM classification is applied to be the objective function in evolutionary process in order to avoid the overfitting problem.

Proposition. (Stability of soft margin SVM classification) Let (x_1, y_1), (x_2, y_2), ..., (x_m, y_m) be the training data where $x_j \in R^N$ is a sample data and $y_i \in \{-1, 1\}$ is its label. Assume $K(\cdot)$ is a bounded kernel, that is $K(x_i, x_j) \le \kappa^2$. The bound with probability at least $1-\delta$ over the sample of size m is

$$R \le R_{emp} + \frac{\kappa^2}{\lambda m} + \left(1 + \frac{2\kappa^2}{\lambda}\right)\sqrt{\frac{\ln(1/\delta)}{2m}}, \tag{11}$$

where R is the risk or generalization error, R_{emp} is called the empirical error, and λ is the regularization parameter of SVM ($\lambda = 1/C$).

The expressions in the right-hand side of (11) are used as the objective function to evaluate parameters of SVM and kernel function. The bound of kernel function (κ^2) can be estimated when the parameters of the kernel are assigned for each individual vector ($\vec{v}$). We presume that a set of suitable parameters should provide a lower bound of risk.

4 Sentiment Classification

We used a dataset of product reviews, which was provided by Bing Liu[1] [16]. This dataset contains sentences used in product reviews collected from the internet and assigned with a sentiment tag: positive or negative. The dataset contains 1,700 sentiment sentences: 1,067 positive and 633 negative sentences. Methods were evaluated by 5-fold cross-validation. The SVM classifiers were trained by using unigrams (single words) as features. The evolutionary strategies were used to find the optimal parameters. The value of τ in evaluation process of these experiments is 1.0. The weights of combination (a_i), and the regularization parameter (C) were real numbers between 0.0 and 10.0. These parameters were inspected within 1,000 generations of ES. The non-negative linear combination kernels for sentiment classification were compared in terms of the average test accuracy. The single kernel functions, i.e., linear kernel, polynomial kernel at different degree, RBF at different scale, and sigmoid kernel are the baselines. The average accuracy values of SVM with single kernel functions are shown in Table 2.

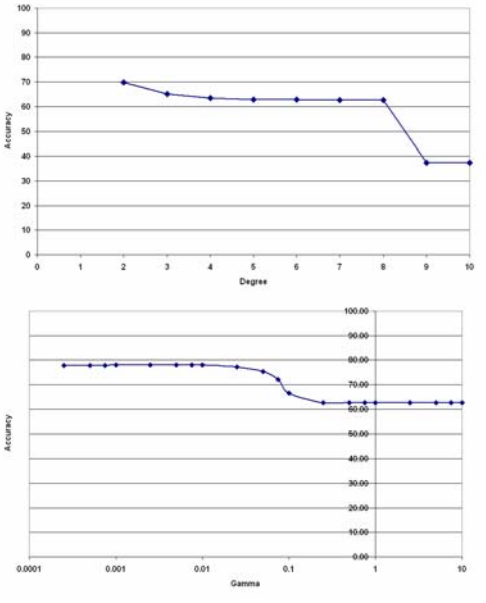
From Table 2, we can see that the linear kernel yielded a good accuracy whereas the sigmoid kernel did not. For polynomial and RBF kernels, the accuracy of sentiment classification is decreased when the degree of polynomial or the width of RBF is increased. The graphs of polynomial and RBF kernels when we varied the parameters are shown in Fig. 1, respectively.

From those graphs, we found that the polynomial at degree 2 yields the result that is better than the other degree. Therefore, we will use degree 2 of polynomial kernel to combine with other kernel functions in the non-negative linear combination kernel.

[1] The dataset is available at http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

Table 2. The Average Accuracy of SVM with Single Kernel Functions

Kernel	Average Accuracy	Kernel	Average Accuracy
Linear	77.7093	Sigmoid	56.0037
Polynomial (d = 2)	69.7711	RBF (γ = 0.0005)	77.8266
Polynomial (d = 3)	65.0086	RBF (γ = 0.005)	78.1216
Polynomial (d = 4)	63.4780	RBF (γ = 0.05)	75.3579
Polynomial (d = 5)	62.7721	RBF (γ = 0.5)	62.6546
Polynomial (d = 6)	62.7723	RBF (γ = 5.0)	62.6546



(a)

(b)

Fig. 1. The accuracy of sentiment classification at different parameters, (a) on Polynomial kernels and (b) on RBF kernels

In the same way, the accuracy of RBF kernel at the width 0.005 is better than those of the other parameters. Hence, the RBF kernel at the width 0.005 will be used for testing on the combined kernels. The average accuracy of SVM with non-negative linear combination kernels is shown in Table. 3.

We tested the SVM with all possible linear combination of these 4 sub-kernels (linear, polynomial, RBF, and sigmoid), and found that the non-negative linear combination of linear and polynomial kernels yields the best result. Although the linear combination of three or four sub-kernels yield more average accuracy on training, the average accuracy on test is not good. This means that the performance of sentiment classification will be improved when the suitable kernels are combined. If we choose

Table 3. The Average Accuracy of SVM with Non-Negative Linear Combination Kernels

Kernel	Average Training Accuracy	Average Test Accuracy
Linear + Poly (2) + RBF (0.005) + Sigmoid	99.2218	69.4782
Linear + Poly(2) + RBF(0.005)	99.7647	70.3580
Linear + Poly(2) + Sigmoid	99.8236	69.7670
Linear + RBF(0.005) + Sigmoid	**99.8530**	70.2391
Poly(2) + RBF(0.005) + Sigmoid	99.7059	68.1829
Linear + RBF(0.005)	98.7355	78.7122
Linear + Poly(2)	98.0881	**79.1828**

the unsuitable kernel, the accuracy may be decreased or the classification model may overfit training data.

Then, the non-negative linear combination of several polynomial kernels at different degree, the non-negative linear combination of several RBF kernels at different scale, and the non-negative linear combination of both polynomial and RBF kernels at different parameters are validated. The number of terms of sub-kernel was fixed as 10. The degree of polynomial sub-kernels are 1, 2, …, 10 for multiple polynomial kernels. For multiple polynomial and RBF kernels, 5 terms of polynomial sub-kernels and 5 terms of RBF sub-kernels are used. The weights of combination (a_i), the widths of RBF kernels (γ_i), and the regularization parameter (C) were real numbers between 0.0 and 10.0. The average accuracy of SVM with non-negative linear combination of multiple polynomial and RBF kernels is shown in Table. 4. These results are compared with the single kernel functions.

Table 4. The Average Accuracy of SVM with Non-Negative Linear Combination of Multiple Polynomial and RBF Kernels

Kernel	Average Training Accuracy	Average Test Accuracy
Linear	88.3533	77.7093
Poly(2)	85.7943	69.7711
RBF(0.005)	89.6472	78.1216
Multiple Polynomial Kernels	98.0881	**79.1828**
Multiple RBF Kernels	95.0294	73.1805
Multiple Polynomial and RBF Kernels	**98.7355**	78.7122

The sentiment classification was tested by the SVM with all 3 kinds of non-negative linear combination on multiple polynomial and RBF kernels. We found that the average accuracy on sentiment classification can be enhanced by the combined kernel.

The non-negative linear combination of multiple polynomial kernels yielded the best result on testing. Although the linear combination of multiple RBF kernels did not yield the best result, its accuracy was better than single RBF kernel. For the combination of both polynomial and RBF kernels, it yielded the best training accuracy, but its accuracy on testing is lower than the combination of multiple polynomial kernels. This means that although we try to avoid the overfitting problem by using the stability objective function in evolutionary process, the overfiitng problem still can be occurred by a more flexible kernel.

5 Conclusion

The non-negative linear combination kernels for SVM were proposed and applied on the sentiment classification problem. This kernel function was more flexible, and there were some adjustable parameters (the weights of combination). The evolutionary strategies were used for adjusting these parameters. In order to avoid the overfitting problem, the stability of soft margin SVM classification was considered to be the objective function in evolutionary process.

The experimental results showed the ability of the proposed method through the average accuracy on 5-fold cross-validation on the sentiment classification problem. The non-negative linear combination kernels yielded the classification results that were better than the single RBF kernels when the suitable kernels were combined. If many sub-kernels were combined, it maybe occur the overfitting problem. However, there are the other combination methods that maybe improve the efficiency of sentiment classification, which we will be investigated in the near future.

Acknowledgement

The authors acknowledge the financial support provided by the Royal Thai Government Scholarship (Mahidol University) and the Thailand Research Fund.

References

1. Kennedy, A., Inkpen, D.: Sentiment Classification of Movie Reviews using Contextual Valence Shifters. Computational Intelligence 22(2), 110–125 (2006)
2. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 79–86 (2002)
3. Li, J., Sun, M.: Experimental Study on Sentiment Classification of Chinese Review using Machine Learning Techniques. In: International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE 2007), Beijing, pp. 393–400 (2007)
4. Mullen, T., Collier, N.: Sentiment Analysis using Support Vector Machines with Divers Information Sources. In: Proceedings of EMNLP (2004)
5. Li, S., Zong, C., Wang, X.: Sentiment Classification through Combining Classifiers with Multiple Feature Sets. In: International Conference on Natural Language Processing and Knowledge Engineering, 2007 (NLP-KE 2007), pp. 135–140 (2007)

6. Okanohara, D., Tsujii, J.: Assigning Polarity Scores to Reviews Using Machine Learning Techniques. In: Dale, R., Wong, K.-F., Su, J., Kwong, O.Y. (eds.) IJCNLP 2005. LNCS (LNAI), vol. 3651, pp. 314–325. Springer, Heidelberg (2005)
7. Vapnik, V.N.: Statistical Learning Theory. John Wiley and Sons, New York (1998)
8. Kecman, V.: Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models. MIT Press, London (2001)
9. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, London (2002)
10. Graf, A., Borer, S.: Normalization in Support Vector Machines. In: Radig, B., Florczyk, S. (eds.) DAGM 2001. LNCS, vol. 2191, pp. 277–282. Springer, Heidelberg (2001)
11. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge (2004)
12. Beyer, H.-G., Schwefel, H.-P.: Evolution strategies: A comprehensive introduction. Natural Computing 1(1), 3–52 (2002)
13. de Doncker, E., Gupta, A., Greenwood, G.: Adaptive Integration Using Evolutionary Strategies. In: Proceedings of 3rd International Conference on High Performance Computing, pp. 94–99 (1996)
14. Fogel, D.B.: Evolutionary Computation: Toward a New Philosophy of Machine Intelligence. IEEE Press, Piscataway (1995)
15. Bousquet, O., Elisseeff, A.: Stability and Generalization. Journal of Machine Learning Research 2, 499–526 (2002)
16. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2004), Seattle, Washington, USA, August 22-25 (2004)