A Matrix Modular SVM Robust to Imbalanced Data for Efficient Visual Concept Detection

Herve Glotin
Sciences and Information Lab.
(LSIS), UMR CNRS
University Sud Toulon-Var,
France
glotin@univ-tln.fr

Zhong-Qiu Zhao
College of Computer Science
and Information Engineering,
Hefei University of Technology,
China
zhongqiuzhao@gmail.com

Jun Gao
College of Computer Science
and Information Engineering,
Hefei University of Technology,
China
gaojun@hfut.edu.cn

Xindong Wu
College of Computer Science
and Information Engineering,
Hefei University of Technology,
China;
Department of Computer
Science, University of
Vermont, USA
xwu@cs.uvm.edu

ABSTRACT

We propose a novel matrix modular support vector Machine (MMSVM) classifier that partitions an image retrieval task into many easier two-class tasks between subsets, each of which is accomplished by a SVM model, and then combines the outputs of the SVM models to produce the final decision. The classifier is tested on ImageClef2009 Photo Annotation, with a comparison with the single SVM model. The experimental results show that our MMSVM model performs well as a classifier in image retrieval, especially in enhancing the classification accuracy for positive samples. We also demonstrate that the MMSVM model has an apparent complementary classification capability to SVM. A good fusion on them might improve the accuracy of image retrieval.

Categories and Subject Descriptors

I.4.8 [**Computing Methodologies**]: IMAGE PROCESSING AND COMPUTER VISION Scene Analysis
; I.5.2 [**Computing Methodologies**]: PATTERN RECOGNITION Design Methodology
; H.3.1 [**Information Systems**]: INFORMATION STORAGE AND RETRIEVAL Content Analysis and Indexing

General Terms

Algorithms, Design, Performance, Reliability, Theory

MIR'10, March 29–31, 2010, Philadelphia, Pennsylvania, USA.

Keywords

Image retrieval, modular classification, SVM, imbalanced data, ImageClef Visual Concept Detection

1. INTRODUCTION

Classification plays an important role in image retrieval, and can greatly affect the retrieval accuracy. In real applications of all sorts of classifiers, their fast and non-local convergent property, good mapping and generalization capability, and adaptive learning behavior are commonly pursued by the users. For simple training data, all these demands are easily met. However, when high complexity becomes an issue, the learning speed and generalization capability of classifiers obviously decrease, and can generally be unsatisfactory.

Some researchers have tried to solve these problems by some output coding methods such as distributed output codes [1] and error-correcting output codes (ECOC) [2]. These methods may improve the generalization capability of the classifiers.

However, distributed output coding requires enough prior analysis and knowledge on the samples, which are absent from most of the conditions; the ECOC method cannot make the size of a training set smaller, and the learning process turns out to be more time-consuming when the number of classes is large.

Nilsson [3] proposed a neural network architecture with several perceptrons and a voting machine, called modular neural networks (MNN). By this technique, a complex supervised learning task can be accomplished by first dividing it into some subtasks and in turn assigning these subtasks to several experts; and then by an integration machine to integrate all results of the experts to produce the solutions for the complex task. Generally speaking, modular classifiers can largely reduce the sizes of the classifiers, and thereby speed up the classifier learning and enhance the generalization capability [3].

So far, many classifiers have been developed, such as KNN, MLP [4], KDA [5], SVM [6], and so on. Among these classifiers, SVM framework is certainly the most powerfull. We experimented some for image retrieval, by which we constructed the 4th best team image retrieval system in ImageCLEF 2008 VCDT [7]. However, it did not always do well for all visual concepts, which may be caused by the imbalanced training data. Actually, image retrieval is generally a 2-class classification problem to distinguish one topic from all other topics. Therefore many more negative training samples than positive ones are provided for training in image retrieval systems. As a result, the whole training data is mightily imbalanced and the training is apt to meet the larger subset, namely the negative samples, but not the positive ones, hence the retrieval accuracy becomes low [8]. A solution scheme using modular neural networks can be found in [9][10], in which the method of pairwise coupling was introduced to decompose a large 2-class classification task into a series of smaller 2-class sub-tasks. Each of these smaller tasks is to distinguish one class from another class, instead of all other classes. So this method can avoid the imbalance of the training data.

Meanwhile, it may occur that the smaller 2-class problems obtained by the above task decomposition are still hard to construct general classifiers. In order to overcome this difficulty, we have proposed to partition these two-class tasks into even smaller two-class subtasks [12]. In this paper we propose a modular classification system called matrix modular support vector machines (MMSVMs) to implement image retrieval tasks.

In this paper, we use Gabor filters as descriptors for images. Gabor filters are directly related to Gabor wavelets, since they can be designed for a number of dilations and rotations. However, in general, expansion is not applied for Gabor wavelets, since it requires the computation of bi-orthogonal wavelets, which may be very time-consuming. Therefore, usually, a filter bank consisting of Gabor filters with various scales and rotations is created. The filters are convolved with the signal, resulting in a so-called Gabor space. This process is closely related to processes in the primary visual cortex. The Gabor space is very useful in image processing applications such as iris recognition, fingerprint recognition and image retrieval. Relations between activations for a specific spatial location are very distinctive between objects in an image. Furthermore, important activations can be extracted from the Gabor space in order to create a sparse object representation.

2. MATRIX MODULAR SVM

In our scheme, we partition each class space into several smaller subspaces. Then our matrix modular classifier architecture divides a complex problem into many much easier subtasks, each of which is to distinguish between one certain subspace and another subspace. These subtasks are then implemented by a series of SVMs, which can make up of a matrix of SVMs. So the proposed MMSVMs mainly contain two parts: a matrix of SVMs and an integration machine (see Figure 1). The input values for the MMSVMs are always presented to the matrix of SVMs, which will yield a matrix of outputs. This matrix of outputs is then fed to the integration machine so that a classification decision can be made.

2.1 Task Decomposition

The retrieval of one topic among K topics is just to distinguish one class from the remaining $K-1$classes, and is a 2-class classification problem. The well-known divide-and-conquer strategy can be used to divide this classification problem into $(K-1)$ smaller 2-class subtasks, each of which is to distinguish between the retrieval topic and one of the other topics; then all the $(K-1)$ pairwise decisions are combined to form the final decision. The detailed decomposition process can be seen in [12]. Thereby, for topic i, we should construct $K-1$ classifiers P_{ij} $(j=1,...,K, j \neq i)$, each of which is to distinguish class i from class j.

The task decomposition is stated in detail as follows. Assuming that χ_k denotes the positive input set for topic k, then

$$\chi_k = \{X_k^l\}_{l=1}^{N_k}, k = 1, 2, ...K$$

where X_k^l is the input values for the positive samples of topic k.

Furthermore, using clustering methods, we can divide the input set of class c_k, χ_k, into D_k subsets as, $\chi_k^d, d = 1, 2, ...D_k$. Then for retrieval topic i we should construct a total of $D_i(D-D_i)$ classifiers, $P_{i^{di}j^{dj}}$, where $D = \sum_{k=1}^K D_k, di = 1, ..., D_i, j = 1, ..., K, j \neq i, dj = 1, ..., D_j$.

2.2 The Matrix of SVMs

For retrieving each topic i, we design two SVM matrices as follows

$$\mathbb{M}_i = (M_{i1}, ..., M_{i(i-1)}, \Phi, M_{i(i+1)}, ..., M_{iK})$$

$$\mathbb{M}_i^{'} = \begin{pmatrix} M_{1i} \\ ... \\ M_{(i-1)i} \\ \Phi \\ M_{(i+1)i} \\ ... \\ M_{Ki} \end{pmatrix}$$

where M_{ij} is a $D_i * D_j$ SVM matrix in charge of distinguishing class i from class j , and

$$M_{ij} = \begin{pmatrix} P_{i^1j^1} & P_{i^1j^2} & ... & P_{i^1j^{D_j}} \\ P_{i^2j^1} & P_{i^2j^2} & ... & P_{i^2j^{D_j}} \\ & ... & ... & \\ P_{i^{D_i}j^1} & P_{i^{D_i}j^2} & ... & P_{i^{D_i}j^{D_j}} \end{pmatrix} = (P_{i^{di}j^{dj}}),$$

$$j = 1, ..., K, j \neq i, di = 1, 2, ..., D_i, dj = 1, 2, ..., D_j.$$

Here $P_{i^{di}j^{dj}}$ is an SVM with only one output node, which generates the output value $o_{i^{di}j^{dj}}$ $(o_{i^{di}j^{dj}} \in [0,1])$. The module of $P_{i^{di}j^{dj}}$ undertakes the subtask of distinguishing the subset χ_i^{di} from that of χ_j^{dj}.

Then the output matrices $\mathbb{O}_i$ and $\mathbb{O}_i^{'}$ yielded by the SVM matrices $\mathbb{M}_i$ and $\mathbb{M}_i^{'}$ respectively, can be described as follows:

$$\mathbb{O}_i = (O_{i1}, ..., O_{i(i-1)}, \Phi, O_{i(i+1)}, ..., O_{iK})$$

$$\mathbb{O}_i^{'} = \begin{pmatrix} O_{1i} \\ ... \\ O_{(i-1)i} \\ \Phi \\ O_{(i+1)i} \\ ... \\ O_{Ki} \end{pmatrix}$$

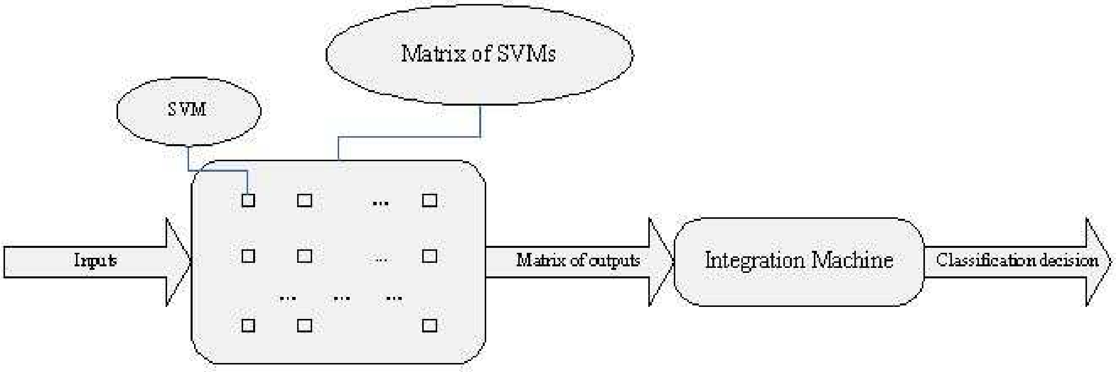

Figure 1: The structure of MMSVMs

where O_{ij} and O_{ji} are the $D_i * D_j$ and $D_j * D_i$ output matrix corresponding to the matrices M_{ij} and M_{ji}, respectively, and

$$O_{ij} = \begin{pmatrix} o_{i^1j^1} & o_{i^1j^2} & \cdots & o_{i^1j^{D_j}} \\ o_{i^2j^1} & o_{i^2j^2} & \cdots & o_{i^2j^{D_j}} \\ & \cdots & \cdots & \\ o_{i^{D_i}j^1} & o_{i^{D_i}j^2} & \cdots & o_{i^{D_i}j^{D_j}} \end{pmatrix} = (o_{i^{di}j^{dj}}),$$

$$j = 1, ..., Kj \neq i, di = 1, 2, ..., D_i, dj = 1, 2, ..., D_j.$$

2.3 Integration Machine

The averaging approach [14] used in general modular classifier systems is to adopt the average of the results from all modules as the basis of the final classification decision. To be used in our proposed MMSVM architecture, our averaging machine is modified as follows. Since the desired outputs of $P_{i^{di}j^{dj}}$ for the samples from χ_i^{di} and those from χ_j^{dj} are set to 1 and 0, respectively, the values of $o_{i^{di}j^{dj}}$ can be regarded as a conditional posterior-probability with which the input belongs to χ_i^{di}, namely

$$Prob(x \in \chi_i^{di} \mid P_{i^{di}j^{dj}}) = o_{i^{di}j^{dj}}.$$

In the matrix M_{ij}, all of the elements which have the desired outputs of 1 for the inputs from χ_i^{di}, denoted by the $\mathcal{P}_i^{di}$ row of the matrix $\mathbb{M}_i$ are in charge of distinguishing χ_i^{di} from all $D - D_i$ subsets of the classes except i. While in the matrix M_{ji}, all elements which have the desired output of 1 for the inputs from χ_j^{dj}, denoted by the $\mathcal{P}_j^{dj}$ row of the matrix $\mathbb{M}_i^{'}$ are in charge of distinguishing χ_j^{dj} from all D_i subsets of the class i. So our averaging approach is to use the average output value of the $\mathcal{P}_i^{di}$ row of $\mathbb{M}_i$ as a new estimate of the posterior-probability with which x belongs to χ_i^{di}:

$$Prob_{average}(x \in \chi_i^{di} \mid \mathcal{P}_i^{di})$$

$$= \frac{1}{D - D_i} \sum_{j=1, j \neq i}^{K} \sum_{dj=1}^{D_j} Prob(x \in \chi_i^{di} \mid P_{i^{di}j^{dj}})$$

$$= \frac{1}{D - D_i} \sum_{j=1, j \neq i}^{K} \sum_{dj=1}^{D_j} o_{i^{di}j^{dj}}$$

where $\mathcal{P}_i^{di}$ denotes the set of the SVMs $P_{i^{di}j^{dj}}$(for all $j = 1, ..., K, j \neq i, dj = 1, ..., D_j$).

We use the average output value of the $\mathcal{P}_j^{dj}$ row of $\mathbb{M}_i^{'}$ as a new estimate of the posterior-probability with which x belongs to χ_j^{dj}:

$$Prob_{average}(x \in \chi_j^{dj} \mid \mathcal{P}_j^{dj}) = \frac{1}{D_i} \sum_{di=1}^{D_i} Prob(x \in \chi_j^{dj} \mid P_{j^{dj}i^{di}})$$

$$= \frac{1}{D_i} \sum_{di=1}^{D_i} o_{j^{dj}i^{di}}$$

where $\mathcal{P}_j^{dj}$ denotes the set of the SVMs $P_{j^{dj}i^{di}}$(for all $di = 1, ..., D_i$).

Then the final decision can be given by:

$$Assign \quad x \longrightarrow Topic \quad i$$

$$if \quad i == \arg max\{Prob_{average}(x \in \chi_k^{dk})\}$$

$$for \quad all\ k = 1, ..., K, dk = 1, ..., D_k.$$

2.4 Desired Outputs

Assuming that a test pattern, x, from χ_k^{dk}, is transmitted to the matrix of SVMs, we will acquire a matrix of outputs O, in which the desired values of $o_{k^{dk}j^{dj}}(j \neq k)$ are all 1 and the desired values of $o_{j^{dj}k^{dk}}$ are all 0, according to the definition of the MMSVM. Thereby, the desired average value of $o_{k^{dk}j^{dj}}$ is 1, while the desired average value of $o_{i^{di}j^{dj}}(i \neq k)$ is absolutely less than that of $o_{k^{dk}j^{dj}}$ because among $D - D_i$ elements of each row $o_{i^{di}j^{dj}}(i \neq k; j = 1, 2; j \neq i; dj = 1, ..., D_j)$, there is at least one element, viz. $o_{i^{di}k^{dk}}$, whose desired value is 0. So the average value of the k^{dk} row of the output matrix must be greater than that of any other row $i^{di}(i = 1, 2; i \neq k; di = 1, ..., D_i)$ if all SVM models perform well.

2.5 Clustering for Subset Divisions

The simple and popular K-means clustering method is used for subset divisions. We use the MSE criterion to determine the number of clusters each class should be divided into. The Figure 2 shows an example of the MSE change with the number of clusters. We performed K-means clustering on the positive and negative sample sets for the topic 'indoor' of VCDT, with the number of clusters varying from 1 to 30. Then the MSE decreases with the increase of the

Table 1: Accuracy rates in % of SVM, MMSVMs and their fusion on ImageCLEF2009 Photo Annotation dataset, where 'AR' denotes the classification accuracy rate on the whole testset, 'PAR' represents the classification accuracy rate for the positive samples, and 'NAR' is the classification accuracy rate for the negative samples.

	SVM	MMSVM	FUSION
AR	88.10	86.68	87.67
PAR	13.07	16.04	14.67
NAR	91.55	88.60	90.70

number of clusters, first sharply and then slightly. An inflection point can be found on the curve for either the positive or negative set, where the number of clusters is equal to 5 and 8, respectively. Thereby we set the number of clusters for the positive and negative sets as 5 and 8, respectively.

3. FAST CLASSIFICATION BY LS SVM

We use the Least Squares Support Vector Machines (LS-SVM) in our MMSVMs system. The SVM [6] first maps the data into a higher dimensional input space by some kernel functions, and then learns a separating hyperspace to maximize the margin. The SVM is typically based on an ε-insensitive cost function, meaning that approximation errors smaller than ε will not increase the cost function value. This results in a quadratic convex optimization problem. The least square support vector machines (LS-SVM) [15] are a reformulation to the standard SVMs which lead to solving linear KKT systems instead, which is quite computationally attractive. Thus, in all our experiments, we will use the LS-SVMlab1.5 from esat.kuleuven. We use the RBF kernel and 10-fold cross validation to tune its sigma.

4. RESULTS AND DISCUSSION

In this section, we compare our MMSVMs with a single SVM on the ImageCLEF2009 Photo Annotation dataset [13]. This dataset provides a training set of 5000 images which are labeled by 53 concepts and a test set of 13000 images. All images may have multiple annotations. Most annotations refer to holistic visual concepts and are annotated at an image-based level. This task poses one main challenge: Can image classifiers scale to a large amount of concepts and data ?

In our experiments, the Gabor filter is used as the visual features [11]. The image is first filtered with a bank of orientation and a scale. The energy in the frequency domain in the corresponding sub-bands is then used as the components of the texture descriptor.

The Table 1 shows the accuracy results of MMSVMs and SVM. It indicates that the MMSVM model has a slightly worse AR than SVM, but that MMSVMs do better on PAR, with an enhancement of about 3 points, worse on NAR with a similar loss.

We plotted in Figure 3 the classification accuracy rate of MMSVMs and SVM by topic. The comparison results are very interesting: for some certain topic, if the PAR performance is improved, then the NAR is decreased. This phenomenon is much consistent with the shortcoming of the SVM algorithm in dealing with an imbalanced dataset. The negative set, with a much larger size than the positive set, attracts most of attention of the SVM learning (which is the same as MLP or other NNs), while the positive set might not be learned sufficiently. However, our MMSVM model decomposes the positive and negative sets into several balanced subsets, so the learning can be averagely shared between positive and negative sets, and then the performance on the positive set is improved while the performance on negative set is decreased. Based on this analysis, we should take into account which of the two costs is larger: the cost of misclassifying positive samples into negative ones and the cost of misclassifying negative samples into positive ones. If the former is larger, then MMSVMs should be selected; otherwise, SVM should be selected.

According to the above analysis, we perform fusion between the MMSVMs and SVM by the following formula:

$$W_i = \frac{N_i}{N},$$

$$Sf_i = W_i * Ss_i + (1 - W_i) * Sm_i,$$

where N denotes the number of all training samples; N_i denotes the number of positive training samples for topic i; Sf_i denotes the fusion score for topic i; Ss_i is the score for topic i obtained from SVM; and Sm_i is the score for topic i obtained from MMSVM.

The fusion results (in Table 1) and top of Figure 3 show that the scores (AR, PAR, NAR) of the fusion are all between those of MMSVMs and SVM. So this fusion could be a good compromise for special requirements. To inspect the characteristics of MMSVMs, we also plotted the count of positive training samples by topic in the bottom of Figure 2. We can see that, for both SVM and MMSVMs, the PAR performance is improved with the increase of the percentage of positive samples on the whole training set, and the NAR performance is improved with the increase of the percentage of the negative samples on the whole training set. So like SVM, MMSVMs can attain better AR if more training samples are provided.

5. CONCLUSION

The proposed MMSVM framework brings a new direction to tackle the imbalance problem in SVM-based image retrieval systems. It performs well as a classifier in image retrieval systems, especially in enhancing the classification accuracy rate for the positive samples. However, the classification accuracy rate for the negative samples is reduced since the learning for the negative set is weakened. So in its applications, we should take into account which of the two costs is larger: the cost of misclassifying positive samples into negative ones and the cost of misclassifying negative samples into positive ones. If the former is larger, then MMSVMs should be selected; otherwise, SVM should be selected.

The simple fusion proposed here between MMSVMs and SVM did not make any improvement. So our further work will be focused on fusion methods by the apparent complementary classification capability for positive and negative samples between MMSVMs and SVM.

6. ACKNOWLEDGMENTS

This work was supported by the French National Agency of Research (ANR-06-MDCA-002), the Research Fund for

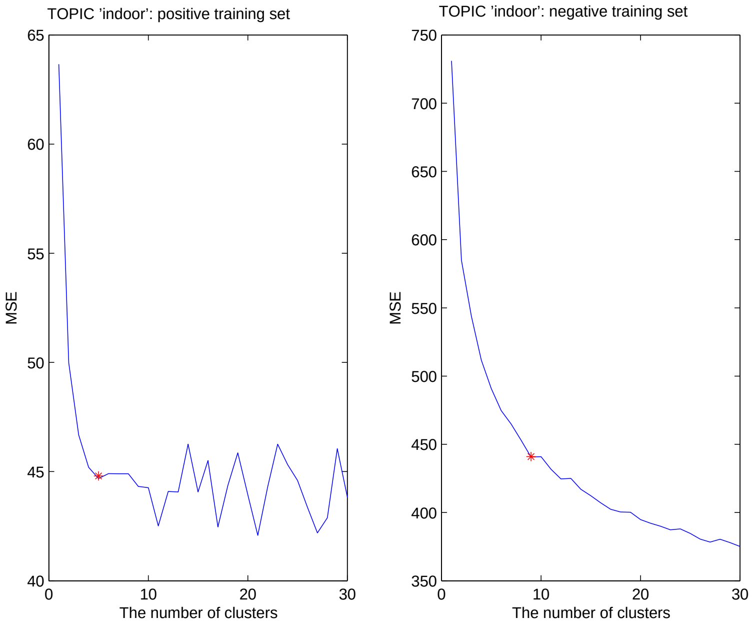

Figure 2: An example of an MSE change with the number of clusters.

the Doctoral Program of Higher Education of China (200803591024), the Fund of HFUT (2007GDBJ054), and the National Natural Science Foundation of China (60875012 and 60828005).

7. REFERENCES

[1] T. J. Sejnowski & C. R. Rosenberg, ”Parallel networks that learn to pronounce english text”, Journal of Complex Systems, vol. 1(1), pp.145-168.

[2] T. G. Dietterich & G. Bakiri, ”Solving Multiclass Learning Problems via Error-Correcting Output Codes,” Journal of Artificial Intelligence Research, vol. 2, pp.263-286, 1995.

[3] N.J. Nilsson : Learning Machines-Foundations of Trainable Pattern-Classifying Systems. New York McGraw-Hill, 1965.

[4] S.K. Pal, S. Mitra : Multilayer perceptron, fuzzy sets, and classification. IEEE Trans. Neural Networks. **3(5)** (1992) 683-697.

[5] G. Baudat, F. Anouar : Generalized discriminant analysis using a kernel approach. Neural Computation. **12** (2000) pp.2385-2404.

[6] V. Vapnik : The nature of statistical learning theory. Springer-Verlag, New York (1995).

[7] H. Glotin, Z.Q. Zhao, S. Ayache : Efficient Image Concept Indexing by Harmonic & Arithmetic Profiles Entropy, IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, November 7-11, (2009).

[8] R. Anand, K. Mehrotra, C.K Mohan and al. : Efficient Classification for Multiclass Problems Using Modular Neural Networks. IEEE Trans. Neural Networks. **6(1)** (1995) pp.117-124.

[9] T. Hastie, R. Tibshirani : Classification by pairwise coupling. Annals of Statistics. **26(2)** (1998) pp.451-471.

[10] E.N. Mayoraz : Multiclass Classification with Pairwise Coupled Neural Networks or Support Vector Machines. Proc. Int.'1 Conf. Artificial Neural Network (ICANN '01). (2001) pp.314-321.

[11] J.G. Daugman, ”Two-dimensional spectral analysis of cortical receptive field profiles”, Vision Res. 20 (10), (1980) pp.847-856

[12] Z.Q. Zhao, D.S. Huang, W. Jia : Palmprint recognition with 2DPCA+PCA based on modular neural networks. Neurocomputing, **71** (2007) pp.448-454.

[13] D. Thomas, H. Allan : The Visual Concept Detection Task in ImageCLEF 2008, Evaluating Systems for Multilingual and Multimodal Information Access – 9th Workshop of the Cross-Language Evaluation Forum. (2008).

[14] L. Xu, A. Krzyzak, C.Y Suen : Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition. IEEE Trans. System, Man, and Cybernetics. **22(3)** (1992) pp.418-433.

[15] J.A.K. Suykens, J. Vandewalle : Least Squares Support Vector Machine Classifiers. Neural Processing Letters. **9** (1999) pp.293-300.

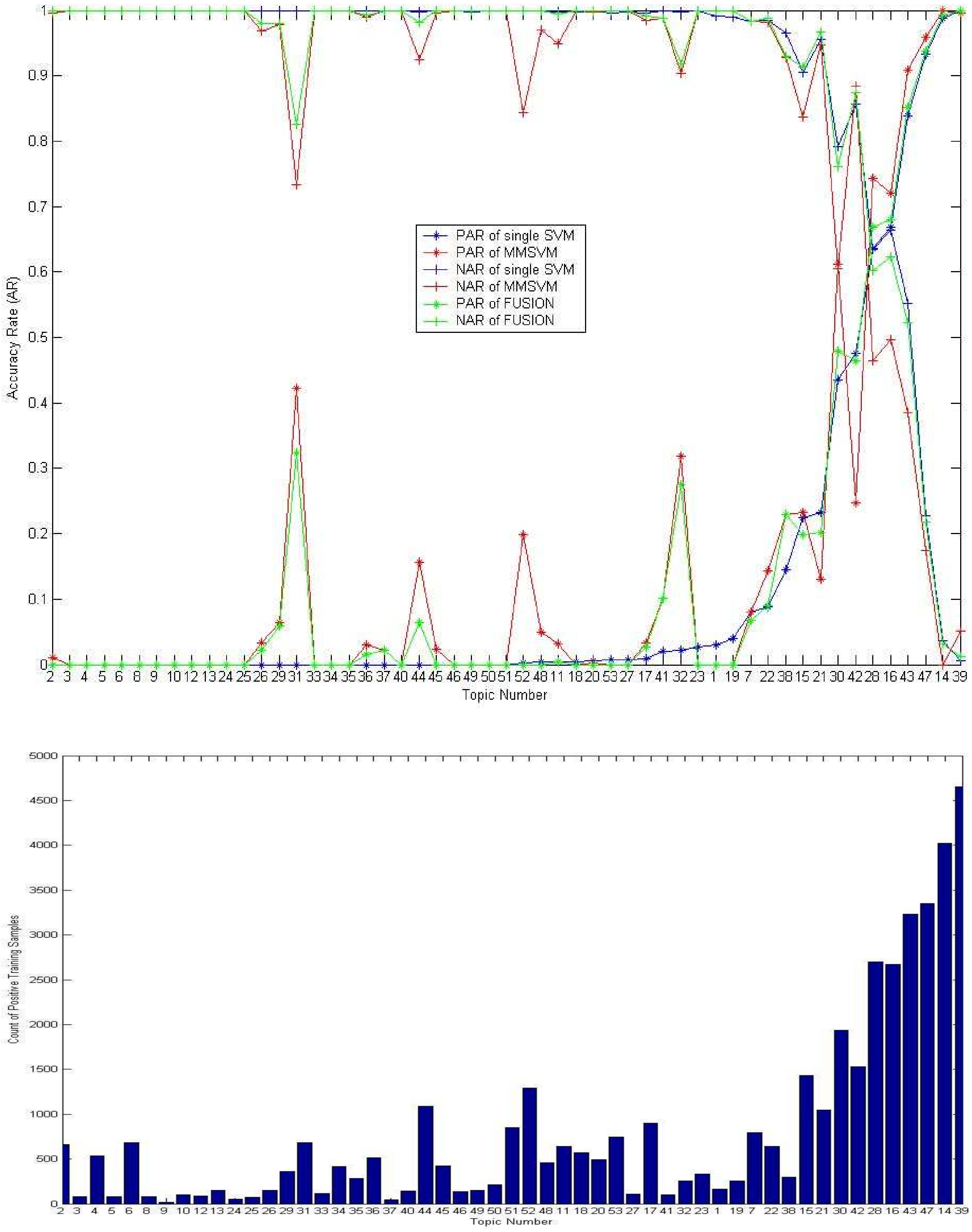

Figure 3: Top: Comparisons of PAR (*) and NAR (+) between MMSVMs and SVM by topic (SVM: blue; MMSVM: red; Fusion: green). Bottom: The counts of positive training samples for the corresponding topic.