

# Prediction of Protein Subcellular Localizations Using Moment Descriptors and Support Vector Machine

Jianyu Shi, Shaowu Zhang, Yan Liang, and Quan Pan

College of Automation, Northwestern Polytechnical University,  
Xi'an, 710072, China

Snake5947@msn.com, {Zhangsw, Liangyan, Quanpan}@nwpu.edu.cn

**Abstract.** As more and more genomes have been discovered in recent years, it is an urgent need to develop a reliable method to predict protein subcellular localization for further function exploration. However many well-known prediction methods based on amino acid composition, have no ability to utilize the information of sequence-order. Here we propose a novel method, named moment descriptor (MD), which can obtain sequence order information in protein sequence without the need of the information of physicochemical properties of amino acids. The presented method first constructs three types of moment descriptors, and then applies multi-class SVM to the Chou's dataset. Through resubstitution, jackknife and independent tests, it is shown that the MD is better than other methods based on various types of extensions of amino acid compositions. Moreover, three multi-class SVMs show similar performance except for the training time.

## 1 Introduction

One of the big challenges in biological field is about structure and function classification and further characterization of protein sequences, as more and more genomes and protein sequences are exploited. It is widely accepted that the subcellular localization of proteins plays a crucial role in predicting protein functions<sup>[1]</sup>. Hence a large number of computation methods have been developed over the last few years. However most of them are based on amino acid composition.

Originally, Nakashima and Nishikawa<sup>[2]</sup> indicated that intracellular and extracellular proteins are significantly different in amino acid composition (AAC). The subsequent studies showed that AAC is closely related to protein subcellular localizations. However, the sequence-order information is ignored in AAC. Hence two sequences, different in function and localization but similar in AAC, may be predicted as the same localization. To utilize the sequence-order information, some novel feature extraction methods have been proposed and may be divided into in the following two categories.

The first category focuses on combining AAC with physicochemical properties of amino acids. Feng and Zhang<sup>[3,4]</sup> considered hydrophobic information and Zp parameters respectively. Chou firstly presented an effective method, named Pseudo

Amino Acid Composition, to predict protein subcellular localization<sup>[5]</sup>. Then Zhou, Cai and Chou further developed this method<sup>[5,7,8,9]</sup>. Pan et al also presented a stochastic signal processing approach<sup>[10]</sup> to predict protein subcellular location based on pseudo amino acid composition.

The other makes the direct extension of AAC. Bhasin and Raghava developed a web server ESLpred for subcellular localization of eukaryotic proteins using dipeptide composition<sup>[11]</sup> and PSI-BLAST. Park and Kanehisa applied compositions of amino acids and amino acid pairs to predict 12-class protein subcellular localizations<sup>[12]</sup>. Cui et al proposed two-segment amino acid composition and developed a tool, named Esub8, to predict protein subcellular localizations in eukaryotic organisms<sup>[13]</sup>.

This paper proposes a novel feature extraction method, named moment descriptor (MD), which takes into account sequence-order information in protein sequence without incorporating physicochemical properties of amino acids. Then MD and multi-class SVMs are used to predict subcellular localizations of proteins.

## 2 Method

### 2.1 Feature Extraction

Without loss of generality, we assume that there are  $N$  protein sequences in the dataset, let  $L_k$  be the length of the  $k$  th sequence  $p_k$ , and  $\alpha_i$  be the  $i$  th element of 20 natural amino acids represented by English letters A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W and Y respectively.

**Amino Acid Composition.** According to amino acid composition, the protein sequence  $p_k$  can be characterized as a 20-D feature vector:

$$AAC_k = [c_1^k, \dots, c_i^k, \dots, c_{20}^k], \quad k = 1, \dots, N \quad (1)$$

where  $c_i^k = n^i / L_k$  is the normalized occurrence frequency of amino acid  $\alpha_i$ , and  $n^i$  is the count of  $\alpha_i$  appearing in sequence  $p_k$ .

However, it is not sufficient to characterize a specific protein sequence only based on  $AAC_k$  because the position of  $\alpha_i$  in protein sequence is not considered. Suppose that we have two protein sequences denoted as  $p_1$  and  $p_2$  with the lengths of 10 and 20, respectively. Amino acid  $\alpha_i$  is occupied at position 2 and 3 in  $p_1$ , and at position 1, 6, 8 and 15 in  $p_2$ . In such case, the information of sequence-order may be needed because  $c_i^1$  equals  $c_i^2$  exactly.

**Moment Descriptor.** Considering the sequence order, we propose a new feature extraction method, called moment descriptor (MD).

Firstly, instead of using above direct definition, we calculate  $c_i^k$  by introducing position indicator  $x_{i,j}^k$  as follows:

$$c_i^k = \frac{1}{L_k} \sum_{j=1}^{L_k} x_{i,j}^k \quad (2)$$

$$x_{i,j}^k = \begin{cases} 1 & \text{if } \alpha_i \text{ is present at position } j \text{ in } p_k \\ 0 & \text{if } \alpha_i \text{ is NOT present at position } j \text{ in } p_k \end{cases} \quad (3)$$

Obviously,  $AAC_k$  in formulation (1) is the sampled statistical mean (raw moment) of position indicator. Hence, we choose formulation (2) as the first MD of protein sequence.

Secondly, considering the position of amino acid  $\alpha_i$  in sequence  $p_k$ , we define a new feature for amino acid  $\alpha_i$

$$m_i^k = \frac{1}{L_k} \sum_{j=1}^{L_k} x_{i,j}^k \cdot j \quad (4)$$

Thus, sequence  $p_k$  can be characterized as a 20-D feature vector

$$AAM_k = [m_1^k, \dots, m_i^k, \dots, m_{20}^k], \quad k = 1, \dots, N \quad (5)$$

where  $m_i^k$  represents mean of position of  $\alpha_i$ .  $AAM_k$  represents the sampled statistical mean of position of amino acids (AAM) in sequence  $p_k$ . We choose it as the second MD.

Here  $AAM_1$  is not equal to  $AAM_2$  in general. However, it is not sufficient just based on AAM to characterize a protein sequence. For example, there may exist two protein sequences  $p_3$  and  $p_4$  with the same length of 10. Amino acid  $\alpha_i$  is occupied at position 8 and 10 in  $p_3$ , and at position 3, 6 and 9 in  $p_4$ . In such case,  $m_i^3$  equals  $m_i^4$  exactly although the positions of amino acid  $\alpha_i$  in both sequences are different.

It may be still not sufficient to characterize a protein sequence based on both AAC and AAM. Suppose there are two protein sequences  $p_5$  and  $p_6$  with the same length of 10. Amino acid  $\alpha_i$  is occupied at position 4 and 6 in  $p_5$ , and at position 3 and 7 in  $p_6$ . It is unfortunate that  $c_i^5$  equals  $c_i^6$ , and  $m_i^5$  equals  $m_i^6$ . Hence it is needed to extract further features from protein sequence.

Thirdly, the sampled variance  $v_i^k$  of position of amino acid  $\alpha_i$  in sequence  $p_k$  is considered:

$$v_i^k = \frac{1}{L_k} \sum_{j=1}^{L_k} (x_{i,j}^k \cdot j - m_i^k)^2 \quad (6)$$

Then, we can obtain a 20-D feature vector

$$AAV_k = [v_1^k, \dots, v_i^k, \dots, v_{20}^k], \quad k = 1, \dots, N \quad (7)$$

where  $v_i^k$  represents the second-order central moment of position of amino acid  $\alpha_i$  in sequence  $p_k$ .  $AAV_k$  represents the sampled statistical variance of position of amino acids (AAV) in sequence. We choose AAV as the third MD of protein sequence.

Eventually, we construct a combined 60-D feature vector for sequence  $p_k$  by combining above three moment descriptors

$$X_k = [AAC_k, AAM_k, AAV_k]^T, \quad k = 1, \dots, N \quad (8)$$

## 2.2 Multi-class SVM

Several classification algorithms have already been applied to protein subcellular localization, such as least Mahalanobis distance<sup>[14]</sup>, neural network<sup>[15]</sup>, covariant discriminant algorithm<sup>[16]</sup>, Markov chain<sup>[17]</sup>, fuzzy k-NN<sup>[18]</sup> and support vector machine<sup>[9,12,19,20]</sup>.

Support vector machine (SVM)<sup>[21]</sup> has been proved to be a fruitful learning machine, especially for classification. Since it was originally designed for binary classification, it is not a straightforward issue to extend binary SVM to multi-class problem. Constructing  $\Omega$ -class SVMs ( $\Omega \geq 2$ ) is an on-going research issue<sup>[22]</sup>.

Basically, there are two kinds of approaches for multi-class SVM. One directly processes all data in one optimization formulation<sup>[23]</sup>. The other decomposes multi-class into a series of binary SVMs, including “One-Versus-Rest” (OVR)<sup>[21]</sup>, “One-Versus-One” (OVO)<sup>[24]</sup>, and DAGSVM<sup>[25]</sup>. Although there are also several sophisticated approaches for multi-class SVM, extensive experiments have shown that OVR, OVO and DAGSVM are practical<sup>[26,27]</sup>.

OVR is probably the earliest approach for multi-class SVM. For  $\Omega$ -class problem, it constructs  $\Omega$  binary SVMs. The  $i$ th SVM is trained with all the positive samples from the  $i$ th class and all negative samples from the other classes. Given a testing sample to classify, all  $\Omega$  SVMs are evaluated, and the testing sample is labeled the class with the largest value of the decision functions.

For a  $\Omega$ -class problem, OVO constructs  $\Omega(\Omega-1)/2$  binary SVMs. During the evaluation, each of the  $\Omega(\Omega-1)/2$  SVMs casts one vote for its most favored class, and finally the class with the most votes wins<sup>[24]</sup>.

Compared with OVO, DAGSVM has the same training process but the different evaluation. During the evaluation, DAGSVM uses a directed acyclic graph (DAG)<sup>[25]</sup> architecture to make a decision. The idea of DAG is easily implemented. Let  $T = 1, 2, \dots, \Omega$  be a list of class labels. When a testing sample is given, DAG first evaluates this sample with the binary SVM, which corresponds to the first and the last elements in list T. If the classifier prefers one of the two classes, then the other one will be eliminated from the list. After each testing, a class label will be excluded. As a result, through  $\Omega-1$  binary SVM evaluations, the last label remaining in the list will be the answer.

Here, SVM software we used is LIBSVM<sup>[26]</sup> which can be freely downloaded from <http://www.csie.ntu.edu.tw/~cjlin/libsvm/> for academic research. We can implement above three methods just through modifying LIBSVM.

### 2.3 Test and Assessment

As mentioned in most papers, the prediction quality is often assessed by the resubstitution, jackknife, and independent dataset tests<sup>[5]</sup>, respectively.

Resubstitution test is used to evaluate the self-consistency of prediction system. During the process of resubstitution test, the subcellular location of each protein in a dataset is predicted by the parameters derived from the same training dataset. Jackknife test is always regarded as the most objective and effective one. During the process of jackknife test, each protein in training dataset is singled out in turn as a testing sample, and the remaining proteins are used as training samples to evaluate the testing sample's class. The quality of independent test indicates the ability of generalization of predictive system in practical application. During the process of independent test, proteins in training dataset are used as training samples and proteins in the independent testing dataset are used as testing samples.

To assess the quality of three tests, the total prediction accuracy and prediction accuracy of each location can be respectively defined as<sup>[18,19,20]</sup>:

$$Total\ accuracy = \frac{1}{N} \sum_{\omega=1}^{\Omega} p(\omega) \quad (9)$$

$$accuracy(\omega) = \frac{p(\omega)}{obs(\omega)} \quad (10)$$

where  $N$  is the total number of sequences,  $\Omega$  is the class number,  $obs(\omega)$  is the number of sequences observed in location  $\omega$  and  $p(\omega)$  is the number of correctly predicted sequences in location  $\omega$

## 3 Experiments and Discussion

Here we train the dataset only with the RBF kernel in all the following experiments. In addition, in order to avoid the domination of the features in greater numeric ranges over those in smaller numeric ranges and numerical difficulties during the calculation, we scale all training data to be in  $[0,1]$  and adjust all testing data with the same transformation accordingly.

### 3.1 Dataset

The training dataset and independent dataset [5] are used to validate the current method. The training dataset consists of 2191 protein sequences, of which 145 are chloroplast, 571 cytoplasm, 34 cytoskeleton, 49 endoplasmic reticulum, 224 extracellular, 25 Golgi apparatus, 37 lysosome, 84 mitochondrial, 272 nuclear, 27 peroxisomal, 699 plasma membrane and 24 vacuoles. The independent dataset consists of 2494 protein sequences, of which 112 are chloroplast proteins, 761 cytoplasm, 19 cytoskeleton, 106 endoplasmic reticulum, 95 extracellular, 4 Golgi apparatus, 31 lysosome, 163 mitochondria, 418 nucleus proteins, 23 peroxisomal, and 762 plasma membrane.

### 3.2 Results of Prediction

In this section, the approach for multi-class SVM is OVO, which is implemented directly by LIBSVM without any changes.

Firstly, in order to show the improvement of position of amino acid is statistically significant, we measure the performance of AAC, AAM and MD in 10-fold cross validation and present mean and standard deviation of prediction accuracies in table 1.

**Table 1.** Mean and standard deviation of prediction accuracies (%) obtained with OVO in 10-fold cross validation

Method	Mean	Standard deviation
AAC	79.78	3.341
AAM	80.14	2.235
MD	83.47	2.698

As shown in Table 1, AAM and MD do improve the classification results with more mean and less standard deviation of prediction accuracy. Maybe, positional bias of amino acid contains more classification information than its compositional bias in protein subcellular localizations. The further exploration is still open.

Then, we apply MD to Chou's dataset and list the prediction accuracies of subcellular localization in Table 2.

As shown in Table 2, the total accuracy in resubstitution, jackknife and independent tests, reaches 99.2%, 79.9% and 85.8% respectively. It seems that classifier suffers from overfitting since the accuracies of resubstitution test are much higher than those on the independent dataset. However, by varying the SVM trade-off parameter from  $2^{-3}$  to  $2^{10}$ , we find that the classifier has no overfitting as a matter of fact in our experiments.

**Table 2.** Prediction accuracies(%) obtained with OVO in resubstitution, jackknife and independent tests, respectively

Location	Resubstitution	Jackknife	Independent
Chloroplast	98.6	75.9	84.8
Cytoplasm	99.7	87.9	88.8
Cytoskeleton	100.0	44.1	94.7
Endoplasmic reticulum	98.0	40.8	84.9
Extracellular	98.2	67.0	80.0
Golgi apparatus	100.0	32.0	50.0
Lysosome	100.0	62.2	96.8
Mitochondrial	96.4	36.9	22.1
Nuclear	99.3	82.0	86.4
Peroxisomal	96.3	25.9	65.2
Plasma membrane	99.7	93.7	97.4
Vacuoles	100.0	29.2	—
Total accuracy	99.2	79.9	85.8

It is also worthy to note that several small groups including cytoskeleton, endoplasmic reticulum, Golgi apparatus, peroxisomal and vacuolar which have 34, 49, 25, 27 and 24 training samples, obtain poor prediction accuracies 44.1%, 40.8%, 32.0%, 25.9% and 29.2% in jackknife test, respectively. Better jackknife prediction may be achieved by increasing the amount of the training samples from updated databases.

Moreover, mitochondrial gets poor predictions 36.9% and 22.1% in both jackknife and independent tests even the amount of its training samples is up to 84. Better prediction may be obtained by subdividing mitochondrial into inner membrane, outer membrane and matrix proteins.

### 3.3 Comparison of Feature Extraction

Here, in order to show the efficiency of MD, we compare it with other methods which make the direct extensions of AAC and extract feature merely from sequence without incorporating physicochemical properties. These methods include the traditional amino acid composition (AAC)<sup>[2]</sup>, amino acid pair/dipeptide composition(AAP)<sup>[11,12]</sup>, and two-segment amino acid composition(2SAAC)<sup>[13]</sup>. We apply above methods respectively to the same Chou's dataset and then compare MD with them. The approach for multi-class SVM is also OVO. The comparison results are presented in Table 3.

**Table 3.** Total accuracies (%) obtained with other methods using OVO in resubstitution, jackknife and independent tests, respectively

Method	Dim	Resubstitution	Jackknife	Independent
AAC	20	92.6	77.2	81.7
AAP	400	98.7	77.8	81.6
2SAAC	40	92.4	79.6	83.8
MD	60	99.2	79.9	85.8

Compared with AAC, AAP and 2SAAC, MD can obtain about 6.6%, 0.5% and 6.8% total accuracy improvements in resubstitution test, about 2.9%, 2.3% and 0.5% total accuracy improvements in jackknife test, and about 3.9%, 4.0% and 1.8% total accuracy improvements in independent test, respectively. These results show that MD is effective and helpful for prediction of protein subcellular localization because it can extract more sequence-order information. In the future, the further improvement will be achieved by incorporating physicochemical properties of amino acids.

### 3.4 Comparison of Multi-class SVMs

In order to make the comparison of three multi-class SVMs mentioned in section 2.2, we also train DAGSVM and OVR based on LIBSVM with some modification of its source codes and present the results in Table 4.

**Table 4.** Total accuracies (%) obtained with DAGSVM, OVR, and OVO in resubstitution, jackknife and independent tests, respectively

Multi-Class SVM	Resubstitution	Jackknife	Independent
MD(DAG)	99.2	80.1	85.6
MD(OVR)	99.2	79.8	85.4
MD(OVO)	99.2	79.9	85.8

We find that OVO, OVR and DAG have very similar classification accuracy and that the difference is mainly focused on the number of support vectors, the training time and the testing time. To validate further these differences, we have run training, resubstitution and independent tests for 10 times, and list the number of support vectors (SV), the maximum (Max) and the minimum (Min) time of them in Table 5, respectively.

**Table 5.** The number of support vectors and the consumed time (second) of DAGSVM, OVR and OVO for training, resubstitution, and independent tests, respectively

Method	Training			Resubstitution		Independent	
	SV	Max	Min	Max	Min	Max	Min
MD(DAG)	1603	2.766	2.765	2.000	2.000	2.219	2.203
MD(OVR)	1686	6.812	6.578	2.328	2.312	2.547	2.422
MD(OVO)	1603	2.765	2.657	2.110	2.094	2.438	2.312

Each binary SVM of OVR is optimized on all the  $N$  training samples although it only requires  $\Omega$  binary SVMs. OVO or DAG has  $\Omega(\Omega-1)/2$  binary SVMs to train, however, the total training time of OVO or DAG is still less because individual binary SVM is trained just on the samples from only two classes. We find that OVR has heavy training computational burden with almost 2.5 times of training time of OVO or DAG in our experiments.

Because the testing time is still dominated by the kernel evaluations, we find that the testing time is almost proportional to the number of support vectors. In addition, we also can see that DAG is really a little faster than OVO on the testing time and needs extra data structure to index the binary SVMs so that it occupies a little bit larger memory than OVO.

As described above, except for the training time, other performance of DAG, OVO and OVR are very similar. Hence, we suggest that DAGSVM and OVO may be more suitable in practical use.

## 4 Conclusion

In this paper, we have developed a novel feature extraction method, called moment descriptor which extract feature merely from sequence without incorporating physicochemical properties, and have applied multi-class SVMs to protein subcellular localization for Chou's protein dataset.



Compared with other methods based on various types of extensions of amino acid compositions, moment descriptor is shown more effectively in representing the protein sequence-order information. Moreover, except for the training time, three types of multi-class SVMs show similar performance. The results show that moment descriptor may be an effective method of feature extraction for protein localization prediction.

**Acknowledgments.** The authors would like to thank Prof. Kuo-Chen Chou (Gordon Life Science Institute, San Diego, CA 92130, USA) for providing the database. This paper was supported, in part, by National Natural Science Foundation of China (No.60372085) and Technological Innovation Foundation of Northwestern Polytechnical University (No. KC02).

## References

1. Feng, Z.P.: An Overview on Predicting Subcellular Location of a Protein. In *Silico Biol.* 2(2002), 0027
2. Nakashima, H., Nishikawa, K.: Discrimination of Intracellular and Extracellular Proteins Using Amino Acid Composition and Residue-Pair Frequencies. *J. Mol. Biol.* 238(1994), 54–61
3. Feng, Z.P., Zhang, C.T.: Prediction of the Subcellular Localization of Prokaryotic Proteins Based on the Hydrophobicity Index of Amino Acids, *Int. J. Biol. Macromol* 28(2001), 255–261
4. Feng, Z.P., Zhang, C.T.: A Graphic Representation of Protein Sequence and Predicting the Subcellular Localizations of Prokaryotic Proteins, *Int. J. Biochem. Cell Biol.* 34(2002), 298–307
5. Chou, K.C.: Prediction of Protein Cellular Attributes Using Pseudo – Amino – Acid – Composition, *Proteins* 43(2001), 246–255
6. Zhou, G.P., Doctor K.: Subcellular Location Prediction of Apoptosis Proteins, *Proteins* 50(2003), 44–48
7. Cai, Y.D. and Chou, K.C.: Nearest Neighbour Algorithm for Predicting Protein Subcellular by Combining Functional Domain Composition and Pseudo Amino Acid Composition, *Biochem. Biophys. Res. Commun.*, 305(2003), 407–411
8. Chou, K.C., Cai, Y.D.: A New Hybrid Approach to Predict Subcellular Localization of Proteins by Incorporating Gene Ontology, *Biochem. Biophys. Res. Commun.* 311(2003), 743–747
9. Chou, K.C., Cai, Y.D.: Using Functional Domain Composition and Support Vector Machines for Prediction of Protein Subcellular Location, *J. Biol. Chem.* 277(2002), 45765–45769
10. Pan, Y.X., Zhang, Z.Z., Guo, Z.M., Feng, G.Y., Huang, Z.D., He, L.: Application of Pseudo Amino Acid Composition for Predicting Protein Subcellular Location: Stochastic Signal Processing Approach, *J. Protein Chem.* 22(2003), 395–402
11. Bhasin, M., Raghava, G.P.S.: ESLpred: SVM-Based Method for Subcellular Localization of Eukaryotic Proteins Using Dipeptide Composition and PSI-BLAST, *Nucleic Acids Res.* 32(2004), W414–W419
12. Park, K.J., Kanehisa, M.: Prediction of Protein Subcellular Locations by Support Vector Machines Using Compositions of Amino Acids and Amino Acid Pairs, *Bioinformatics* 19 (2003), 1656–1663

13. Cui, Q., Jiang, T., Liu, B., Ma, S.: Esub8: A Novel Tool to Predict Protein Subcellular Localizations in Eukaryotic Organisms, *BMC Bioinformatics* 5(2004), 66–72
14. Chou, K.C.: A Novel Approach to Predicting Protein Structural Classes in a (20-1)-D Amino Acid Composition Space, *Proteins* 21(1995), 319–344
15. Reinhardt, A., Hubbard, T.: Using Neural Networks for Prediction of the Subcellular Localization of Proteins, *Nucleic Acids Res.* 26(1998), 2230–2236
16. Chou, K.C., Elrod, D.: Protein Subcellular Localization Prediction, *Protein Eng.* 12(1999), 107–118
17. Yuan, Z.: Prediction of protein subcellular localizations using Markov chain models, *FEBS Lett.* 451(1999), 23–26
18. Huang, Y., Li, Y.D.: Prediction of protein subcellular locations using fuzzy k-NN method, *Bioinformatics* 20(2001), 21–28
19. Hua, S.J., Sun, Z.R.: Support Vector Machine Approach for Protein Subcellular Localization Prediction, *Bioinformatics* 17(2001), 721–728
20. Zhang, S.W., Pan, Q., Zhang, H.C., Shao, Z.C., Shi, J.Y.: Prediction Protein Homologer Types by Pseudo Amino Acid Composition: Approached with an Improved Feature Extraction and Naive Bayes Feature Fusion, *Amino Acid*(2006), in press
21. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York(1998)
22. Bredensteiner, E., Bennet, K.: Multicategory Classification by Support Vector Machines. *Comput. Optim. Appl.* 12(1999), 53–79
23. Crammer, K., Singer, Y.: On the Algorithmic Implementation of Multiclass Kernel-Based Vector Machines, *J. Mach. Learn. Res.* 2(2001), 265–292
24. Kreßel, U.: Pairwise Classification and Support Vector Machines, In Schölkopf, B., Burges, C.J., Smola, A.J.(eds): *Advances in Kernel Methods: Support Vector Learnings*, Cambridge, MA, MIT Press ,(1999) 255–268
25. Platt, J., Cristianini, N., Shawe-Taylor, J.: Large Margin DAGs for Multiclass Classification, In Solla, S.A., Leen, T.K., Muller, K.-R.(eds): *Advances in Neural Information Processing Systems*, Vol. 12. Press (2000) 547–553
26. Hsu, C., Lin, C.J.: A Comparison of Methods for Multi-Class Support Vector Machines, *IEEE. T. Neural Networks* 13(2002), 415–425
27. Rifin, R. and Klautau, A.: In defense of one-vs-all classification, *J. Mach. Learn. Res.* 5(2004), 101–141