社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

大規模汎用パターン認識モデル CombNET-II のための非線形大分類ネットワーク

マウリシオ・クグレ† 宮谷　俊行† 黒柳　奨† 岩田　彰†

† 名古屋工業大学大学院　工学研究科情報工学専攻　〒 466-8555 名古屋市御器所町
E-mail: †mauricio@kugler.com, ††tottsi@mars.elcom.nitech.ac.jp, †††{bw,iwata}@nitech.ac.jp

あらまし　我々の提案する大規模汎用パターン認識モデル CombNET-II においては詳細識別ネットワーク (branch network) の数が増えた場合には、大分類ネットワーク (stem network) の分類性能がボトルネックとなっていた。この原因として stem network の分離平面が参照ベクトルによるボロノイ境界と成っていることがあげられる。そこで本稿では大分類領域間の複雑な境界を学習により形成し、かつ各領域における所属サンプル数のバランスを維持可能な非線形大分類ニューラルネットワークを CombNET-II に適用することを提案し、その有効性について報告する。
キーワード　大規模パターン分類問題, 分割統治法, 大分類ネットワーク, 逐次分類

Non-linear gating network for the large scale classification model CombNET-II

Mauricio KUGLER†, Toshiyuki MIYATANI†,

Susumu KUROYANAGI†, and Akira IWATA†

† The authors are with the Department of Computer Science & Engineering, Nagoya Institute of Technology, Gokiso-cho, Showa-ku, Nagoya, 466-8555, Japan.
E-mail: †mauricio@kugler.com, ††tottsi@mars.elcom.nitech.ac.jp, †††{bw,iwata}@nitech.ac.jp

Abstract　The linear gating classifier (stem network) of the large scale model CombNET-II had been always the limitation factor for increasing the number of expert classifiers (branch networks). The linear boundaries between its clusters cause a rapidly decrease of the performance with the increase of the number of clusters and, consequently, impairing the whole structure performance. This work proposes the use of a non-linear classifier to learn the complex boundaries between the clusters, increasing the gating performance while keeping the balanced split of samples produced by the original sequential clustering algorithm. The experiments showed that, for some problems, the proposed model overcomes even the monolithic classifier.
Key words　large scale classification problems, divide-and-conquer, gating network, sequential clustering

1. Introduction

The large scale classification model CombNET-II, proposed by Hotta *et al.* [1], is a divide-and-conquer based method able to deal with databases of thousands of categories. It presented several good results in Chinese character recognition (Kanji) and some other specific applications. In its basic form, the CombNET-II is composed by a gating network (stem network) and many expert networks (branch networks). The stem network is a modified Vector Quantization (VQ) based sequential clustering called Self Growing Algorithm (SGA), while the branch networks are basically independent Multilayer Perceptrons (MLP). Essentially, the stem network is used to divide the feature space in R Voronoi subspaces, each of them becoming the training data for each independent MLP.

For large scale problems, however, the use of raw data on the stem network training causes the classes to be shattered among the clusters, creating very unbalanced problems for the branch networks, which also becomes to have too many classes. These two factors can make the branch network training very complex and slow. A solution for this is the

use of the average of each class samples on the SGA algorithm training. This procedure, besides reducing the stem network training time, avoids that the classes are split over the clusters, reducing the number of classes per cluster and improving the balance of samples of different classes inside each branch network.

However, the averaged data does not represents thoroughly the real data, specially for complex distributions. If the real training samples were applied on the stem network trained with the averaged samples, a bad performance could be expected. This problem tends to get worst as the number of clusters increase, as the feature space learned by each branch network starts to differs more and more from the feature space represented by the correspondent stem cluster. Clearly, there is a compromise between the stem and the branch networks performance.

This paper proposes a new solution that breaks this compromise, increasing the stem network performance while keeping the advantages of the use of averaged data. An independent MLP is used to represent the complex boundaries between the clusters generated by the use of averaged data, increasing the stem network performance without interfering on the balance of the branch networks feature space.

Non-linear algorithms had already been used as gating networks for large scale model. Collobert, Bengio and Bengio [2], [3] used a MLP gating in their large scale model. However, in their approach, the training data splitting started randomly, being iteratively redefined based on the expert networks performance, requiring the gating to be retrained on each iteration, making the procedure very time consuming. Waizumi *et al.* [4] presented a new rough classification network for large scale models based on a hierarchy of Learning Vector Quantization (LVQ) neural networks, however, without any result of the application of their gating network in a complete large scale model. The method proposed in this paper uses a hybrid gating, in which the non-linear algorithm learns the data splitting generated by the unsupervised clustering algorithm.

The organization of the paper goes as follows: a more detailed revision of CombNET-II is presented in section 2 and section 3 introduces the proposed model, its modifications and new characteristics. Section 4 presents experiments with the new model and some comparisons with the original linear gating and section 5 closes the paper with the discussion of the results and

2. Large Scale Classifier CombNET-II

As explained before, the CombNET-II is composed by a gating network, called "stem" network, and many expert networks, called "branch" networks, with its basic structure

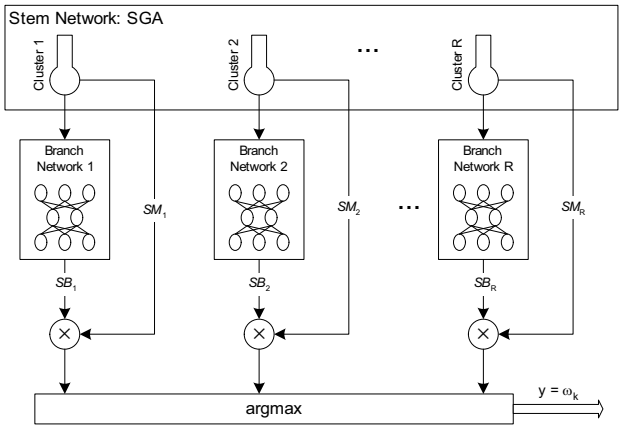

図 1 CombNET-II structure

shown in Figure 1.

The stem network is a VQ based sequential clustering with some modifications to control the balance between the clusters, which replaces the Self Organizing Map (SOM) used in the original CombNET [5]. Its basic algorithm is shown in Figure 2, in which ℓ is the number of samples, R is the current number of clusters, $\mathbf{x}_i$ is the i^{th} sample, $\mathbf{m}_j$ is the j^{th} cluster reference vector, Θ_{s} is the similarity threshold, Θ_{p} is the inner potential threshold, h_j is the j^{th} cluster inner potential and $sim\left(\mathbf{x}_i, \mathbf{m}_j\right)$ represents the similarity measurement between the i^{th} sample and the j^{th} cluster (usually, the normalized dot product).

The branch networks are independent MLP networks, and have their classification results weighted by the stem network scores as described in equation (1).

$$y = \omega_k \left| S_k = \arg\max_j \left(SM_j^{\gamma} \cdot SB_j^{1-\gamma} \right) \right. \tag{1}$$

where:

$$SM_j = sim\left(\mathbf{x}, \mathbf{m}_j\right) = \frac{\langle \mathbf{x}, \mathbf{m}_j \rangle}{|\mathbf{x}|\,|\mathbf{m}_j|} \tag{2}$$

SB_j is the maximal score among the output neurons of the j^{th} branch network and ω_k is the k^{th} possible category, $k = 1, \ldots, K$. The exponent γ is a weighting parameter ($0 \leqq \gamma \leqq 1$) that dictates which network (stem or branch) plays the major role in the classification. The flowchart of the complete CombNET-II training procedure is shown in Figure 3.

As can be seen in the algorithm shown in Figure 2, there is no control about the number of classes in each cluster or the balance of the classes inside each cluster. With the use of averaged data, this control is not necessary (considering that the original classes are nearly balanced), as the control of the clusters size already regulates the number of classes on it. The stem network classification result, however, tends to become poor. The next section presents the proposed strategy

Make $\mathbf{m}_1 = \mathbf{x}_1$, $h_1 = 1$ and $R = 1$
for $i \in \{2 \dots \ell\}$
  Find $\mathbf{m}_c$ so that:
    $sim(\mathbf{x}_i, \mathbf{m}_c) = \max_j [sim(\mathbf{x}_i, \mathbf{m}_j)]$
  if $sim(\mathbf{x}_i, \mathbf{m}_c) > \Theta_s$
    $R = R + 1$, $\mathbf{m}_R = \mathbf{x}_i$, $h_R = 1$
  else
    $\mathbf{m}_c = \mathbf{m}_c \cup \mathbf{x}_i$
    if $h_c > \Theta_p$
      Divide $\mathbf{m}_c$ in $\mathbf{m}'_c$ and $\mathbf{m}_{R+1}$ so that:
        $|h_c - h_{R+1}| \leqq 1$
  end if
end for
do Update the clusters (with necessary divisions)
until No significant changes in any clusters

図 2　Self Growing Algorithm (SGA)

for improving the stem network performance trained with averaged data.

3. Proposed Model

Instead of changing the clustering result in order to search for different data splits that could improve the stem network result without sacrificing the branch networks performance, this paper propose the use of a non-linear algorithm to learn the complex boundaries between the clusters generated by the use of averaged data on the SGA training. The training data of this algorithm would be the same data used to train the original stem network, but with the samples categories relabeled to the cluster they belongs to. The flowchart of the proposed method is shown in Figure 4.

At first, the SGA algorithm is trained using the averaged samples $\bar{\mathbf{x}}_k$ of each k^{th} class. With the obtained clustering result, the raw data is split using the cluster belonging information by:

$$\mathbf{x}_i \in \mathbf{m}_j \leftrightarrow \bar{\mathbf{x}}_k \in \mathbf{m}_j \tag{3}$$

The samples belonging to cluster $\mathbf{m}_j$ are used to train the j^{th} branch network. Independently, the raw data is also relabeled by the clustering information by:

$$y'_i = j \leftrightarrow [y_i = k,\ \ \bar{\mathbf{x}}_k \in \mathbf{m}_j] \tag{4}$$

where y_i and y'_i are respectively the original and the new label of the i^{th} sample. The raw data relabeled by equation (4) is the training data for the non-linear gating network.

The new classification problem created for the non-linear stem network have the following characteristics: relatively small number of categories, large number of samples per class and good balance between the classes. These characteristics suggest the use of MLP as the stem non-linear algorithm [6], and that was the choice for the proposed method. To avoid misunderstandings between the branch MLPs and the stem network MLP, the last one will be abbreviated S-MLP.

Using the One-versus-Rest (OvR) output encoding, the number of outputs neurons on the S-MLP becomes equal to the number of clusters generated by the SGA. The S-MLP training is independent of the branch MLP networks and can be made in parallel, as shown in the flowchart in Figure 4. Moreover, it can be retrained with different parameters, which can be optimized taking in account the complete CombNET-II recognition rate performance, without requiring the branch networks retraining.

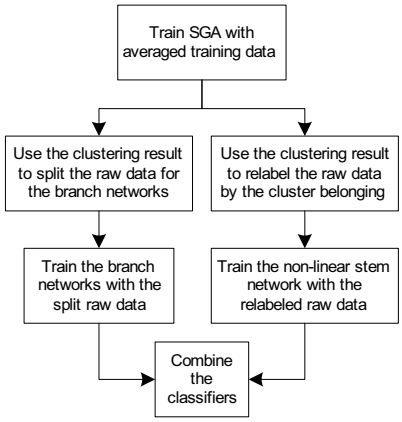

図 4　Proposed model flowchart

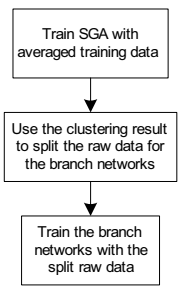

図 3　CombNET-II flowchart

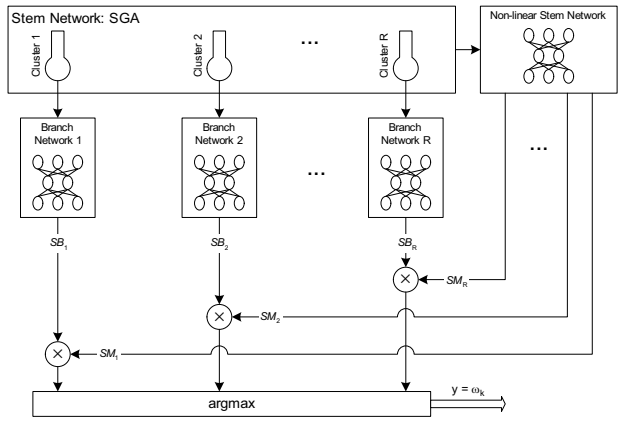

図 5　Proposed model structure

表 1 *Alphabet* database SGA training parameters and results

Number of Clusters	Similarity Threshold	Inner Potential Threshold	Normalized Std. Dev.
1	-1	30	0
2	-1	15	0.0780
3	0.1	14	0.0666
4	-1	8	0.0888
5	0.45	8	0.2507
6	-1	6	0.1192
7	0.75	6	0.1314
8	0.7	5	0.1424

On the recognition stage, the clustering result of the SGA is not needed any more. The unknown sample is inputted directly on the S-MLP and, instead of the linear stem network similarity, each SM_j will correspond to the S-MLP j^{th} output neuron result, to be multiplied by the correspondent SB_j value in equation (1).

The final structure of the proposed model is shown diagrammatically in Figure 5.

4. Experiments

The experiments showed in this paper intend to verify the proposed model's performance gain for the cases where the SGA is trained with averaged data. That is the case of large scale problems, for which is impracticable to train the stem network with raw data. Therefore, even the databases used in this paper experiments can not be considered large (and so allowing the stem network to be trained with raw data), they can properly represent the problem of using averaged data in the SGA. The medium size experiment permitted to extensively optimize the parameters, given a better idea of the models behavior.

The same linear stem network and branch networks were used for both models, choosing the best branch parameter for each case. The MLP neural networks (both branch MLP and S-MLP) were trained until the error was smaller than 10^{-4} or the iteration number exceeds 10^3, with learning rate equal to 0.9, momentum 0.1 and sigmoidal activation function slope 0.1, while the number of hidden neurons and the γ parameter were optimized (testing several values) for each experiment realization.

Two databases were used two verify the performance of the proposed model: *Alphabet* and *Isolet*. The following section present the results for each problem.

4.1 JEITA-HP *Alphabet* Database

The *Alphabet* database consists of the roman alphabet characters subset of the JEITA-HP database (注1) dataset A.

(注1) : Available under request from http://tsc.jeita.or.jp/ /TSC/COMMS/4_IT/Recog/database/jeitahp/index.html

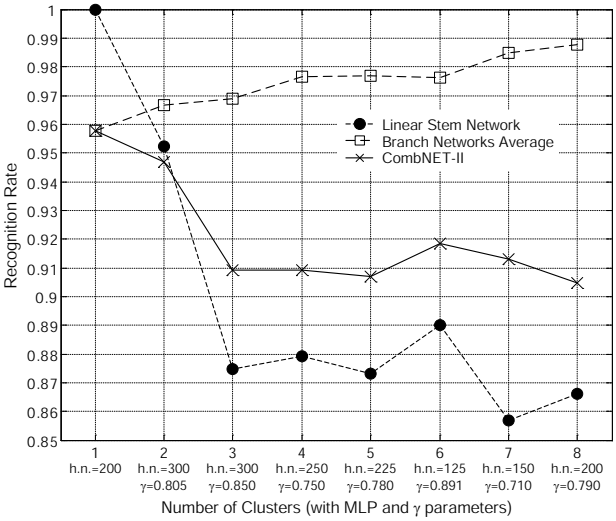

図 6 CombNET-II with linear stem network recognition rate result for the *Alphabet* database

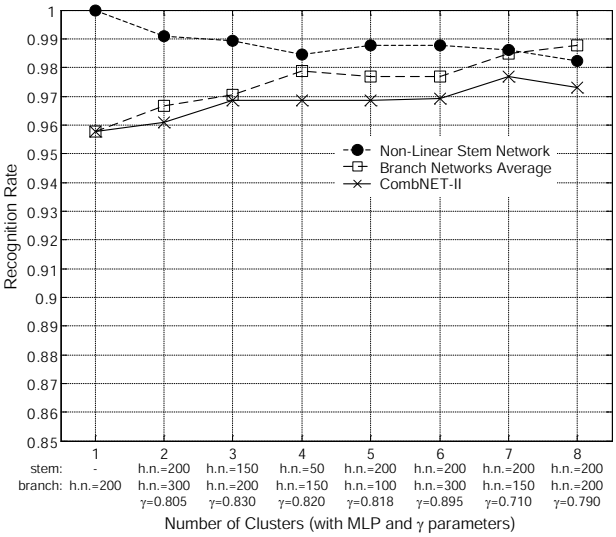

図 7 CombNET-II with non-linear stem network recognition rate result for the *Alphabet* database

The first 200 samples of each character from A to Z were selected for the experiment, with 150 for training (3900 samples) and 50 for testing (1300 samples). The raw characters, composed by 64x64 binary values representing black and white dots, were preprocessed by a Local Line Direction (LLD) feature extraction method [7], which generated 256 features. Each sample vector was normalized to a unitary maximal feature value and zero feature mean. This vector normalization improves the SGA normalized dot product similarity measurement efficiency.

The stem network was trained with several parameters in order to obtain increasing number of clusters, with the best possible balance of number of classes between them. For balanced cluster, the non-optimal procedure of using the same set of parameters for all the branches gives acceptable results. Table 1 shows the parameters used to train each stem

表 2 *Isolet* database SGA training parameters and results

Number of Clusters	Similarity Threshold	Inner Potential Threshold	Normalized Std. Dev.
1	-1	30	0
2	-1	15	0.2176
3	-1	10	0.0666
4	-1	12	0.1986
5	-1	7	0.1609
6	-1	6	0.1884

network and the obtained results. The normalized standard deviation of the number of classes per cluster gives a measurement of how balanced a clustering is, being equal to zero when all the clusters have the same number of classes and increases as the clusters begin to be more and more unbalanced.

Figures 6 and 7 depict the results for the traditional (linear gating) and the proposed (non-linear gating) models respectively, showing the variation of the stem and branch networks and the whole structure recognition rate with the increase of the number of clusters in which the data is divided. Under the abscissae, the optimized parameters for each number of clusters are shown.

There was a significative improvement in the error rate by the use of the non-linear gating. The clear dependence of the linear gating performance with the balance between the clusters (indicated by the normalized standard deviation of the number of classes in each cluster in Table 1) is no longer observed, besides a great improvement on the recognition rate of high number of clusters. The CombNET-II error rate with the linear gating (with 2 clusters or more) was between 5.3% for 2 clusters and 9.5% for 8 clusters, while the non-linear gating presented error rates between 2.3% for 7 clusters and 3.9% for 2 clusters, an improvement between 26.1% and 73.4%.

4.2 UCI *Isolet* Database

The *Isolet* database, obtained from the UCI repository [8], contains 26 categories representing spoken names (in English) of each letter of the alphabet. Each letter was spoken twice by each of the 30 speakers, totalizing 7800 samples (3 of them are missing), divided in 6238 samples for training and 1559 for testing, with 617 features per sample.

Table 2 shows the parameters used to train each stem network and the obtained results. Figures 8 and 9 depict the results for the traditional and the proposed models respectively. Again, the optimized parameters for each number of clusters are shown under the abscissae.

The proposed model also presented a significant improvement in the error rate of the *Isolet* database, specially for high number of clusters. The CombNET-II error rate with

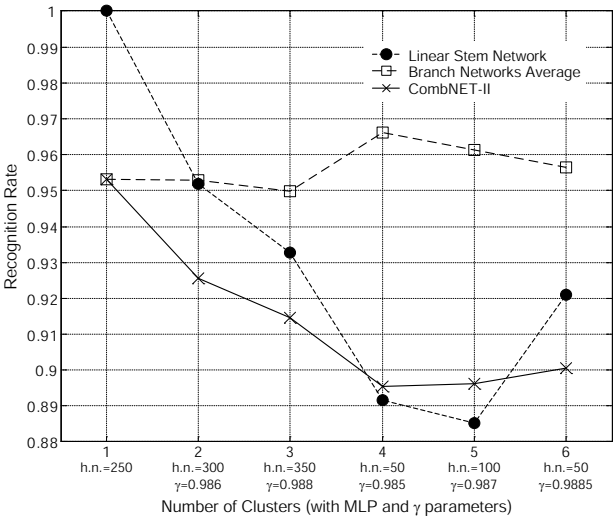

図 8 CombNET-II with linear stem network recognition rate result for the *Isolet* database

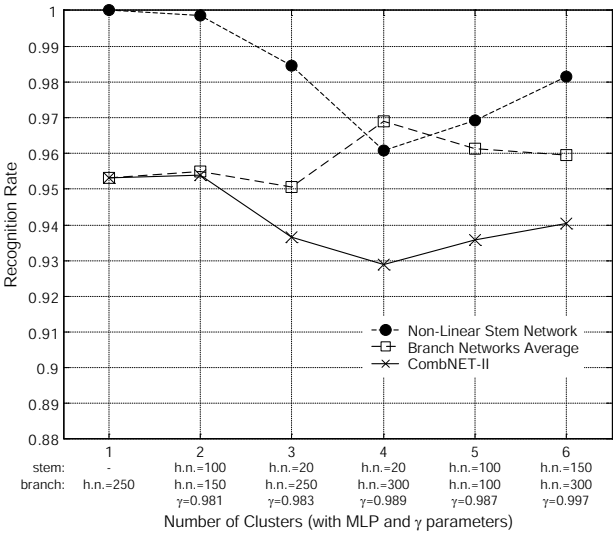

図 9 CombNET-II with non-linear stem network recognition rate result for the *Isolet* database

the linear gating (with 2 clusters or more) was between 7.4% for 2 clusters and 10.5% for 4 clusters, while the non-linear gating presented error rates between 4.6% for 2 clusters and 7.1% for 4 clusters, an improvement between 25.6% and 40%.

5. Discussion and Conclusions

The results shown in Figures 6 to 9 confirm the superiority of the proposed method. As expected, a considerable higher performance of the stem network was obtained by the use of a non-linear classification algorithm. The error rate of the stem network was reduced in between 80.6% to 91.4% for the *Alphabet* database and between 63.9% to 97.3% for the *Isolet* database, in comparison with the linear stem network. Furthermore, this had a consequent error rate reduction on the CombNET-II up to 73.4% for the *Alphabet* database and 40% for the *Isolet* database.

The independence of the non-linear stem network, due to the use of the same clustering information used to split the data for the branch networks, make the proposed model very flexible and easy to implement and train. However, for very large databases, the training time for the stem network can be a bottleneck on the system, as it uses the whole raw training data (even it being relabeled for a small number of categories).

Future works include methods for reducing the training time of the non-linear stem network by reducing the size of the training data and the use of other kinds of (dis)similarity measurements on the stem network for performance improvement, as the use of non-linear algorithms makes free the use of non-limited measures (e.g. Euclidean distance), as the similarity measure is no longer necessary to calculate the SM scores for the CombNET-II output. Other types of non-linear algorithms instead of the S-MLP could also be evaluated.

Acknowledgments

The first author is supported by the Ministry of Education, Culture, Sports, Science and Technology, Government of Japan, and also by a grant from the Hori Information Science Promotion Foundation, Japan.

文　　献

[1] K. Hotta, A. Iwata, H. Matsuo, and N. Susumura, "Large scale neural network CombNET-II," IEICE Transactions on Information & Systems, vol.J75-D-II, no.3, pp.545–553, March 1992.

[2] R. Collobert, S. Bengio, and Y. Bengio, "A parallel mixture of SVMs for very large scale problems," Neural Computation, vol.14, no.5, pp.1105–1114, May 2002.

[3] R. Collobert, S. Bengio, and Y. Bengio, "Scaling large learning problems with hard parallel mixtures," International Journal on Pattern Recognition and Artificial Intelligence, vol.17, no.3, pp.349–365, 2003.

[4] Y. Waizumi, N. Kato, K. Saruta, and Y. Nemoto, "High speed and high accuracy rough classification for handwritten characters using hierarchical learning vector quantization," IEICE Transactions on Information & Systems, vol.E83-D, no.6, pp.1282–1290, June 2000.

[5] A. Iwata, T. Touma, H. Matsuo, and N. Suzumura, "Large scale 4 layered neural network "CombNET"," IEICE Transactions on Information & Systems, vol.J73-D-II, no.8, pp.1261–1267, August 1990.

[6] S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd ed., Prentice Hall, New Jersey, 1998.

[7] H. Kawajiri, T. Yoshikawa, J. Tanaka, A.S.Nugroho, and A. Iwata, "Handwritten numeric character recognition for facsimile auto-dialing by large scale neural network CombNET-II," Proceedings of the 4th International Conference on Engineering Application of Neural Networks, Gibraltar, pp.40–46, June 1998.

[8] C. Blake and C. Merz, "UCI repository of machine learning databases." Irvine, CA: University of California, Department of Information and Computer Science, 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html.