AN OUTPUT CODING APPROACH FOR KNOWLEDGE INCREASABLE ARTIFICIAL NEURAL NETWORK

Hua Huang Siwei Luo

(Computer Science & Technology Department, Northern Jiaotong University, Beijing, China)

E-mail: idapro@sina.com.cn Zip Code: 100044

Abstract: How to inherit the learned knowledge of existing neural networks without destroying their structure and functionality is a difficult problem. In this paper, we proposed an output coding approach for building such a system, which fully utilized the information gained from the component neural units. By coding the neural outputs, a neural network becomes a self-contained system. For a given pattern, such a neural network can correctly recognize or reject it or point out it is similar to patterns it had learned. Such Information is useful for further decision. Experiments demonstrate it a good approach for building KIANN system. This is meaningful for utilizing the learned knowledge of existing neural networks and for large scale parallel processing.

Keywords: knowledge increase, neural network, coding theory

Ever since neural network method was proposed as an approach of machine intelligence, it has been applied to various fields and gained great success. However, the scale and complexity of existing neural network system are still very limited, especially when compared to that of human brains. This limits its application on problems of giant pattern set or large classes. Besides, the structure of existing neural network is inextensible, resulting in its limited knowledge capacity. What's more, no effective incremental learning algorithm made things even worse. For such reasons, researchers worldwide have put great emphasis on this subject, as [1] has mentioned. Various models and methods were proposed to scale up the learning algorithm or improve the processing capability, including composite neural network [2] and hierarchical mixture of experts (HME)[3]. In contrary to incremental learning, such practice paved a novel way for large pattern set processing by combining existing neural network units. The key point of these methods lies in the combining strategy, which greatly affects the effectiveness of such system. Besides, the extensibility of these methods is also confined by the decomposing strategy of problems, which determined its unsuitability for processing giant pattern sets.

In this paper, we present an output coding method for KIANN system. By coding the output of neural network, a neural network can not only output class labels, but also provide a measurable distance to its nearest class for a given input pattern, which can be employed for further decision. Based on such neural network unit, KIANN system can be built, which can inherit the learned knowledge of each component neural unit without destroying their structure and functionality.

The paper is organized as follows. The first section introduces some practice based on "divide and conquer" strategy, and points out the limitations. With the following section give some coding theory with neural network. In the third part, KIANN system model based on the output coded neural network (OCNN) units is presented. The fourth section shows experiments and result, followed with out conclusion in the last part.

1. PRACTICE OF MODULAR STRATEGY

Although neural network can approximate any function of any complexity on theory, the samples needed and time requirements for training such a highly complicated network increase dramatically. That's why modular strategy was adopted. By dividing a complex problem into smaller sub ones, the benefits gained are not only simplicity, but also efficiency. Besides, it also provides the potentiality for extension and for large scale parallel processing. The typical structure for such purpose is combining classifiers. In such models, modules usually have same functionality, with the aim of improving accuracy by ensemble method. The composite neural network model [2] is a typical one. It consists of M component modules together with a gating network, as shown in figure 1. The main drawback of such network lies in its inextensibility for the reason that the gating network is responsible for processing all the input patterns. Although as a whole the system can inherit the knowledge of each component, these are mainly used for improving accuracy rather than for processing large pattern sets. Each component only provides a class label. There is no gradation or confidence information on the output for further processing. Thus all post processing is solely based on the output itself. If each unit can give an annotation on its output, then much better conclusion can be made in the next decision procedure.

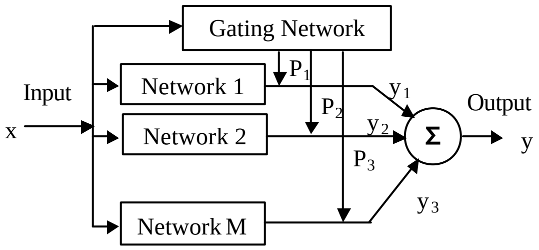

Figure 1 Composite network with M components

2. CODING NEURAL NETWORK OUTPUTS

As above-mentioned, the appraisal of a neural unit to its output may be of great help for further processing. If we code the neural network output with any pair of code words having a distance larger than a threshold, then the distance between actual output on recognition and standard code words may provide such measurement.

Coding theory was originally proposed for solving communication problems. By coding the information with redundancy, message transmission error because of noise can be detected or corrected. Error-correcting codes are just for this purpose. The original motivation for encoding multiple classifiers using error-correcting codes is based on the idea of modeling the prediction task as a communication problem, in which class information is transmitted over a channel. Errors introduced into the process arise from various aspects of the learning algorithm, including features selected, distortion in the input data and finite training samples.

For a given feed-forward neural network with more than one output nodes, conventional assignment methods for each class can be performed in the following ways. One is to have only one output node "ON" and all the remaining nodes "OFF". The number of output nodes is equivalent to the number of classes. Another possible way is to assign a class number to each of the class and use its binary representation as the label of the class. Training is carried out with pairs of patterns and their associated code word. Because the minimum Hamming distance of such methods is only one or two, one class can be easily recognized as another for reason of distortion or noises. From error-correcting theory, we know that a matrix designed to have d bits error-correct capability implies that there is a minimum Hamming distance 2d+1 between any pair of code words. Suppose the minimum Hamming distance of any pair of code word is at least 2d+1, then it is possible to correct a received pattern having fewer than d bits in error by assigning the pattern to the code word closest in Hamming distance, given that each bit is transmitted independently. Practices [4,5] have proven the feasibility and effectiveness of such scheme.

3.KIANN SYSTEM

By coding the output of neural network, the robustness and fault-tolerance of neural network can be improved. This has been proven in many practices, as shown in [6,7]. However, most of such methods are confined to improve robustness or accuracy of neural networks. In fact, the coded output can be utilized for serving other purpose. This is what we will address in this paper.

From another point of view, the coded output of a neural network can provide extra information other than the output itself about the given pattern based on Hamming distance, such as whether the network can handle the input pattern properly or giving comments on the input based on the knowledge it has learned. This information is of great importance for system expansion. If the network can handle the input pattern, its output result may be taken as the final result, otherwise, some other network may be chosen for processing it. Thus scalable neural network system may be constructed with the whole system inherit the learned knowledge of each

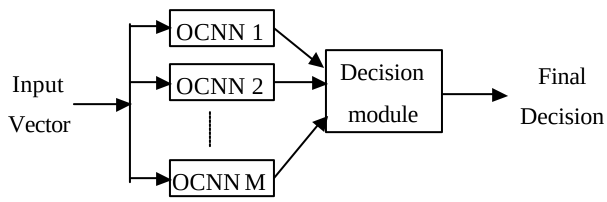

Figure 2 KIANN Model Based on OCNN Units

component unit.

3.1 CODING METHODS

Before constructing such a scalable neural network system, coding problem must be solved because it is the base of all. According to coding theory, many methods can be utilized for error correction or detection. However, considering the specialty and purpose of coding the outputs of neural network, the coding can be simplified. What we need is just minimum distance between each pair of code words. Therefore, we must firstly design the minimum distance between each pair of code word. Let's denote the distance as 2d+1. Then for a given pattern, the distance of 0 to d between the network output and those code words represents the similarity degree. Considering the fact that minimum distance is the No. 1 issue and no correction is required, BCH coding method may be suitable for serving such propose.

Assuming that we have to send n bits digital data $X_1 \in Z^n, Z = \{0,1\}$, an extra k bit of redundancy data is also needed for error-correction or detection. So the total data became $X \in Z^m, m = n + k$. By BCH coding theory, X is determined as

$$X = X_1 \otimes W \qquad (1)$$

where W is an $n \times m$ generator matrix and $\otimes$ is multiplication function in over Galois field (GF(2)). By BCH coding theory, generator matrix is expressed as

$$W = [E_{m-k} \quad W_1] \qquad (2)$$

where E_j is a j dimensional unit matrix, W_1 is part of W. The relation between the redundancy data length k and the original data length m is described as follow

$$k = \log_2(m) * t \qquad (3)$$

where t is the error bits can be corrected. For detailed information, [8] can be referenced.

3.2 KIANN SYSTEM MODEL

By coding the neural outputs, a neural network becomes a self-contained system. For a given pattern, such a neural network can either correctly recognize or reject it. Besides, it can also point out similar patterns it has learned according to the Hamming distance between its output and standard class labels. Such Information can be utilized for building KIANN system of OCNN units. Figure 2 shows the structure of KIANN model.

Because of coded output, each component neural network unit can have three behaviors for a given input pattern: (1) declare a recognition of it with confidence; (2) claim that it was unrecognizable but point out it may belong to some class having been learned; (3) reject it as a unseen pattern. There may be two levels of rejections: ① the output value is between a and $1-a$, where a is a small threshold value. ② distance between the output and any standard code word is larger than a threshold value, which may be usually set to 1 or 2. If there is only one module in the system, the output of the network would be either accepted as the final result or rejected as unrecognizable for condition (1) and (3). For condition (2), we can either reject or accept the output of the network in accordance with the Hamming distance between the output label and the nearest code word. For example, we may identify the input pattern as belonging to the class that has a Hamming distance smaller than a threshold and reject it as unrecognizable for all others. However, things become complicated for a system consisting multiple OCNN units. Several conditions may appear. (1) All modules declare a rejection of the input pattern, then an absolutely rejection would be taken as the final decision. (2) Recognition claimed by only one unit with all others making a rejection declaration, then the output of the unit who claimed the recognition would be taken as the final result of the system. (3) Multiple component units make a claim of recognition, under such circumstance we can extract the training samples of those classes that the input pattern is claimed to belong to and make a dynamic classification. Since that the number of possible classed is greatly reduced, then fast training algorithm or non-neural network approaches can be utilized. In our experiments, we used Support Vector Machines.

The decision module can make decisions as the following steps:

(1) If multiple units claim recognition of the input pattern, extract the training samples of those classes that have a zero Hamming distance

between the unit output label and the predefined code words and go to step (4), otherwise go to step (2).

(2) If all but one unit declares a rejection of the input pattern, accept the output of the one that made the recognition declaration as the final result, otherwise go to step (3).

(3) If all units reject the input pattern and none proposes a similar class that has a Hamming distance of 1 between the output label and the predefined code words, reject the input as a not learned pattern. Otherwise extract the training samples of those similar classes and go to step (4).

(4) Perform dynamic classification for the given input pattern over the extracted training samples and take its result as the final result.

4. EMPIRICAL RESULTS

Based on the model mentioned above, we constructed a KIANN system with OCNN units for printed digits recognition. For simplicity, The 64x64 dot matrix of digit 0 to 9 were directly extracted from Windows and 16x16 binary graphs were gained after processing. These digits were then divided into three groups, 0-3 the first, 4-6 the second and 7-9 the third, each with different degree of random noise. The dot matrixes were directly input to train the OCNNs. The minimum Hamming distance between each pair of code words was set to 5.

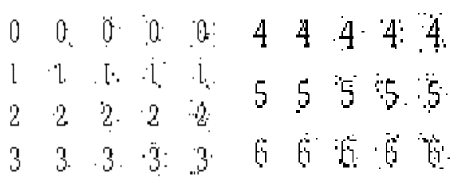

Figure 3 Parts of training samples

The test set consists 100 samples, with 10 variations for each digit. The OCNNs are added to the KIANN sequentially. Before an OCNN unit is added to the system, the samples belonging to this neural unit are successfully rejected by the system. However, the system claims a successfully recognition to these samples after the insertion of the module, which reveals that the KIANN system has successfully inherited the learned knowledge of its component neural unit.

5. CONCLUSION

The OCNN approach for KIANN proposed in this paper is proven to be a complete success, which makes the learned knowledge of each neural unit be successfully inherited by the KIANN system. Each neural unit plays active role in the final decision making procedure. Experiments demonstrate that such a scheme can be successfully utilized for system expansion. This is of great importance for knowledge increasable neural network system and can be used for giant pattern sets.

Acknowledgement:

The National Nature Science Foundation of China supports this paper under contract No. (69973002).

[1] Dietterich,T.G, “Machine learning research: four current directions”, AI Magazine 18:97-136,1997

[2] G. Deco and D. Obradovic, An Information Theoretic Approach to Neural Computing. Springer, pp: 219-224, 1996

[3] Yan Pingfan, Zhang Changshui, Artificial Neural Network and Simulated Evolutionary Computation, Tsinghua University Press, pp: 268-280, 2000

[4] Robert E. Schapire, Using output codes to boost multiclass learning problems, Proceedings of the Fourteenth International Conference In Machine Learning, pp: 313-321, 1997.

[5] Al-Mashouq, Noise tolerance of output-coded neural net, Digital Signal Processing Workshop Proceedings, pp: 442-445, 1996

[6] T.G. Dietterich and G. Bakiri, Solving multi-class learning problems via error-correcting output codes, Journal of AI Research, Vol. 2, pp.263-286, 1995

[7]. Ito, H and Yagi T., Fault tolerant design using error-correcting code for multilayer neural networks, Proceedings, The IEEE International Workshop on Defect and Fault Tolerance in VLSI Systems, IEEE, pp.177-184, 1994

[8] J.H. Van Lint, Introduction to coding theory, Press of science, pp: 95-102, 1988