Available online at www.sciencedirect.com

SCIENCE DIRECT®

Pattern Recognition 39 (2006) 747–760

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

RBF-based neurodynamic nearest neighbor classification in real pattern space

Mehmet K. Muezzinoglu*, Jacek M. Zurada

Electrical and Computer Engineering Department, University of Louisville, Louisville, KY 40292, USA

Received 21 January 2005; received in revised form 20 October 2005; accepted 20 October 2005

Abstract

Superposition of radial basis functions centered at given prototype patterns constitutes one of the most suitable energy forms for gradient systems that perform nearest neighbor classification with real-valued static prototypes. It is shown in this paper that a continuous-time dynamical neural network model, employing a radial basis function and a sigmoid multi-layer perceptron sub-networks, is capable of maximizing such an energy form locally, thus performing almost perfectly nearest neighbor classification, when initiated by a distorted pattern. The proposed design scheme allows for explicit representation of prototype patterns as network parameters, as well as augmenting additional or forgetting existing memory patterns. The dynamical classification scheme implemented by the network eliminates all comparisons, which are the vital steps of the conventional nearest neighbor classification process. The performance of the proposed network model is demonstrated on binary and gray-scale image reconstruction applications.

Keywords: Neurodynamics; Nearest neighbor classification; Associative memory; Radial basis functions

1. Introduction

Nearest neighbor pattern classification is the problem of evaluating the association map

$$f(\mathbf{z}) = \arg \min_{\mathbf{y}\in M} d(\mathbf{z}, \mathbf{y}) \tag{1}$$

defined on a pattern space $\mathbb{P}$, where $M \subseteq \mathbb{P}$ is a finite set of prototype patterns and $d(\cdot, \cdot)$ is a metric on $\mathbb{P}$. A system that calculates Eq. (1) for given M and $\mathbf{z}$, called the Nearest Neighbor Classifier (NNC), is the focus of the design problem in this paper.

A straightforward way of evaluating exactly Eq. (1) for any given instance ($\mathbf{z} \in \mathbb{P}$, $M \subseteq \mathbb{P}$) requires computation of an array of $m = |M|$ distances from $\mathbf{z}$ to each $\mathbf{y} \in M$, then obtaining the index of the minimum element through comparisons, and finally extracting the pattern $\mathbf{y}^* \in M$ associated with the resulting index. This three-stage procedure can be implemented easily on digital computers and it necessitates m evaluations of the metric $d(\cdot, \cdot)$ together with $m - 1$ pairwise comparisons among the distance array. In addition to its high computational requirements, the method also requires the prototype patterns be stored explicitly in the physical memory of the system to be extracted after the comparison phase. A particular case of the problem for $\mathbb{P} = \{0, 1\}^n$, named the nearest codeword problem, has been reported in Ref. [1] as NP-complete. This result may be extended to an arbitrary $\mathbb{P}$.

1.1. *Conventional neural associative memory and its limitations*

In the development years of neural network theory, a single layer of n discrete neurons within a feedback loop was facilitated to retrieve binary patterns from their distorted versions in Ref. [2]. This concept has triggered an enormous interest in analysis and design of finite-state recurrent neural networks, since these neurodynamical systems exploit

* Corresponding author. Tel.: +1 502 852 3165; fax: +1 502 852 3940.
E-mail addresses: kerem@ci.louisville.edu (M.K. Muezzinoglu), jmzura02@louisville.edu (J.M. Zurada).

0031-3203/$30.00 © 2005 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2005.10.026

the memory storage and recovery property, providing a motivation towards explaining the biological associative memory by collective operation of basic computational units. It is possibly for this reason that, artificial neural systems that demonstrate the pattern reconstruction property, even partially for some $\mathbf{z} \in \mathbb{P}$, have been accepted conceptually as associative memories in the related literature. However, there is still a significant gap from engineering viewpoint between NNC, i.e. the ideal associative memory, and neural associative systems, as detailed below. Based on this fact, we view the iterative map realized by a recurrent network model from its initial state vector to the steady state as an approximation of (1), setting formally the NNC as the objective on auto-associative memory design.

The way Hopfield Associative Memory (HAM) [2] operates has indeed three major advantages over the conventional implementation described above:

(1) The computation of the associative map is left to the autonomous dynamics of the network, so no comparison is performed explicitly throughout the process.
(2) The network structure is independent of the number m of prototypes to be stored.[1]
(3) Convergence to a fixed point is guaranteed in at most n^2 steps, when the network has symmetric weight parameters and is operated in asynchronous mode [3].

On the other hand, HAM performs so poor in evaluating (1) that it hardly qualifies as a binary NNC. One of the several shortages that one faces in HAM design is that all elements of an *arbitrary* M might not be introduced as fixed points of the network, irrespective of the design strategy adopted. In fact, complete storage of M in HAM is possible only when the prototype patterns satisfy certain rules. These constraints vary from one design method to other, but, as a common consequence, they impose upper bounds on the cardinality of M. For example, for the classical outer product rule proposed in Ref. [2], such a constraint has been reported in Ref. [9] as $m < 0.14n$. The absolute bound for symmetric asynchronous HAM has been estimated experimentally as $m < 1.5n$ in the work [10], which proposes also a design method to attain this bound.

Another drawback of HAM is the inevitable occurrence of spurious fixed points introduced to its state space by the design method adopted. Unfortunately, neither the number nor the locations of spurious memories can be found in advance. Almost all design methods proposed for HAMs introduce spurious memories, though a set of very strict conditions on M has been derived for a particular design method that ensures a spurious-memory-free state space in Ref. [11].

Because HAM works in the binary space due to the hard-limiter-type activations of the computational units in the network, it cannot handle non-binary patterns. This restriction can be relaxed towards a grid-like finite space by introducing alternative quantizers, such as a multilevel step [12] or a complex signum function [13]. Most of the recurrent network models found in the literature beyond HAM to approximate (1) are still finite-state machines. A significant exception is the M-model [14], whose state space is constrained within the hypercube $[-1, 1]^n$ due to saturated linear activation functions. However, the design procedure, namely the eigen-structure method proposed for this model still aims at storage and recall of binary patterns, i.e. the vertices of the hypercube.

Most of the limitations mentioned above for HAM in approximating (1) apply in general for any recurrent network with a fixed structure independent of the cardinality of M. They can be explained by an energy function approach to the network dynamics: A *fixed network model* may be associated only to a *fixed form of energy function* defined over its state space, which is minimized locally along trajectories generated by the network dynamics. In particular, as proven in Ref. [3], the energy function associated to the HAM model has a quadratic form, and hence the network is able to recall a point only if it is designated as a local minimum to this energy form by a design method. In other words, a given set M of n-dimensional prototype vectors cannot be stored altogether as fixed points of HAM, unless there exists a pair $(\mathbf{Q}, \mathbf{c})$ such that the quadratic $Q(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{c}^T\mathbf{x}$ has a local minimum at each element of M. Similar restrictions apply for energy forms associated with other recurrent models with a fixed structure. Therefore, no fixed network model is capable of handling all possible (unrestricted) prototype combinations $M \subseteq \mathbb{P}$ in general. For instance, there exists no single-layer second-order recurrent network that has distinct fixed points at the three prototypes $\{[0\,0]^T, [0\,1]^T, [1\,0]^T\}$, excluding $[1\,1]^T$.

1.2. Adaptive network structure and proposed model

In fact, all physical memories, including the biological ones, have an adaptive structure to store arbitrary patterns as it is essential for an information system to adapt itself to the information to be processed. This adaptation may be in the form of including new (or by removing existing) processing units. In particular, in order to achieve perfect storage and recall of an arbitrary M by a neural associative memory, the design method must not only adjust the network parameters, but also the network structure. This constitutes the motivation of this work.

In this paper, we propose a continuous-time gradient neural network model with adaptive structure that qualifies as an NNC. The dynamics of the model is defined in the bounded state space $[0, 1]^n$ such that it maximizes an associated scalar energy function, which is in the form of a sum of Gaussian

[1] This would have been a really valuable property from information theoretic point of view, if there would have existed a design method capable of mapping an arbitrary M as the fixed point set of the network, which utilizes $n^2 + n$ parameters.