Neural Networks 24 (2011) 247–256

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Receptive field optimisation and supervision of a fuzzy spiking neural network

Cornelius Glackin *, Liam Maguire, Liam McDaid, Heather Sayers

University of Ulster, Faculty of Engineering, School of Computing and Intelligent Systems, Magee Campus, Londonderry, BT48 7JL, Northern Ireland, United Kingdom

ARTICLE INFO

Article history:
Received 8 December 2008
Received in revised form 19 November 2010
Accepted 30 November 2010

Keywords:
Clustering methods
Receptive fields
Evolutionary algorithms
Spiking neural network
Supervised learning

ABSTRACT

This paper presents a supervised training algorithm that implements fuzzy reasoning on a spiking neural network. Neuron selectivity is facilitated using receptive fields that enable individual neurons to be responsive to certain spike train firing rates and behave in a similar manner as fuzzy membership functions. The connectivity of the hidden and output layers in the fuzzy spiking neural network (FSNN) is representative of a fuzzy rule base. Fuzzy C-Means clustering is utilised to produce clusters that represent the antecedent part of the fuzzy rule base that aid classification of the feature data. Suitable cluster widths are determined using two strategies; subjective thresholding and evolutionary thresholding respectively. The former technique typically results in compact solutions in terms of the number of neurons, and is shown to be particularly suited to small data sets. In the latter technique a pool of cluster candidates is generated using Fuzzy C-Means clustering and then a genetic algorithm is employed to select the most suitable clusters and to specify cluster widths. In both scenarios, the network is supervised but learning only occurs locally as in the biological case. The advantages and disadvantages of the network topology for the Fisher Iris and Wisconsin Breast Cancer benchmark classification tasks are demonstrated and directions of current and future work are discussed.

1. Introduction

The history of neural network research is characterised by a progressively greater emphasis paid to biological plausibility. The evolution of neuron modelling with regard to the complexity of computational units can be classified into three distinct generations (Maass, 1997). The third generation of neuron modelling (spiking neurons) is based on the realisation that the precise mechanism by which biological neurons encode and process information is poorly understood. In particular, biological neurons communicate using action potentials also known as spikes or pulses. The spatio-temporal distribution of spikes in biological neurons is believed to 'hold the key' to understanding the brain's neural code (DeWeese & Zador, 2006).

There exists a multitude of spiking neuron models that can be employed in spiking neural networks (SNNs). The models range from the computationally efficient on the one hand to the biologically accurate on the other (Izhikevich, 2004); the former are typically of the integrate-and-fire variety and the latter are of the Hodgkin–Huxley type. All the models in this range exploit time as a resource in their computations but vary significantly in the number and kinds of neuro-computational features that they can model (Izhikevich, 2004). The extensive amount and variety of neuron models exist in acknowledgement of the fact that there is a trade-off between the individual complexity of spiking neurons and computational intensity.

In addition to the variety of neuron models, biological neurons can have two different roles to play in the flow of information within neural circuits. These two roles are excitatory and inhibitory respectively. Excitatory neurons are responsible for relaying information whereas inhibitory neurons locally regulate the activity of excitatory neurons. There is also experimental evidence to suggest that the interaction between these two types of neuron is responsible for synchronisation of neuron firing in the cortex (Börgers & Kopell, 2003). Ongoing physiological experiments continue to illuminate the underlying processes responsible for the complex dynamics of biological neurons.

The degree to which these complex dynamics are modelled in turn limits the size and computational power of SNNs. Therefore, it is imperative to determine which biological features improve computational capability whilst enabling efficient description of neuron dynamics. Ultimately neuro-computing seeks to implement learning in a human fashion. In any kind of algorithm where human expertise is implicit, fuzzy IF-THEN rules can provide a language for describing this expertise (Zadeh, 1965). In this paper, the rationale for the distribution of biologically-inspired computational elements is prescribed by the implementation of fuzzy IF-THEN rules. This rationale will demonstrate how strictly biological models of neurons, synapses and learning can be assembled in a network topology using fuzzy reasoning. Two benchmark classification datasets are used to demonstrate the capabilities of the

* Corresponding author. Tel.: +44 2871 375249; fax: +44 2871 375470.
E-mail address: c.glackin@ulster.ac.uk (C. Glackin).

0893-6080/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2010.11.008

topology. The benchmarks are the Fisher Iris and Wisconsin Breast Cancer datasets.

In Section 2, unsupervised and supervised learning methods, dynamic synapses and receptive fields (RF) are reviewed. A brief discussion of how fuzzy reasoning can provide a basis for structuring the network topology in such a way that these various elements are suitably utilised follows. Section 3 introduces a generic network topology and outlines the specific models and algorithms used to implement fuzzy reasoning. Section 4 is concerned with pre-processing of benchmark data, Fuzzy C-Means clustering and thresholding to determine cluster size. Experimental results and remarks for the complex non-linear Iris classification problem using a subjective cluster thresholding approach are presented in Section 5, and results from the evolutionary optimisation of the thresholding technique for the Wisconsin Breast Cancer dataset are presented in Section 6. A discussion of generalisation and the main contribution of the work are outlined in Section 7, and lastly conclusions and future research directions are presented in Section 8.

2. Review

In this section, unsupervised and supervised learning methods, dynamic synapses and RFs are reviewed. Modelling synapses is an essential aspect of accurate representation of real neurons, and one of the key mechanisms to reproducing the plethora of neuro-computational features in SNNs. Learning in all generations of neural networks involves the changing of synaptic weights in the network in order for the network to 'learn' some input–output mapping. From a biologically plausible point of view synaptic modification in spiking neurons should be based on the temporal relationship between pre- and post-synaptic neurons, in accordance with Hebbian principles (Hebb, 1949). In fact, Hebbian learning and its ability to induce long-term potentiation (LTP) or depression (LTD) provides the basis for most forms of learning in SNNs. Hebbian learning gains great computational power from the fact that it is a local mechanism for synaptic modification but also suffers from global stability problems as a consequence (Abbott & Nelson, 2000).

2.1. Unsupervised learning

There are several learning algorithms that can be used to evoke LTP or LTD of synaptic weights. STDP (Bi & Poo, 1998; Markram, Lübke, 1997) is a re-establishment of Hebb's causality condition (Hebb, 1949) of strengthening the weight associated with a presynaptic neuron only if there is a high probability that it caused a postsynaptic spike, and weakening the connection otherwise. More specifically, the STDP learning rule dictates that long-term strengthening of the synaptic efficacy occurs when a pre-synaptic spike (AP) precedes a post-synaptic one. Synaptic weakening occurs with the reverse temporal order of pre and postsynaptic spikes. The stability of STDP can be ensured by placing limits in the strengths of individual synapses and a multiplicative form of the rule introduces an adaptive aspect to learning, resulting in progressively smaller weight updates as learning progresses.

Bienenstock, Cooper and Munro's model (BCM) (Bienenstock, Cooper, & Munro, 1982) compares correlated pre- and post-synaptic firing rates to a threshold in order to decide whether to induce LTP or LTD. The threshold slides as a function of the post-synaptic firing rate in order to stabilise the model. Despite criticism for its lack of biological basis (Abbott & Nelson, 2000), BCM has been demonstrated to be related to STDP (Izhikevich & Desai, 2003). In particular, by restricting the number of pairings of pre- and post-synaptic spikes included in the STDP rule, the BCM rule can be emulated using STDP.

BCM and STDP are of course unsupervised learning algorithms, and as such they do not obviously lend themselves to applications requiring a specific goal definition, since this requires supervised learning.

2.2. Supervised learning

There are several methodologies to date for implementing supervised learning in SNNs:

- SpikeProp (Gradient Estimation) (Bohte, Kok, & La Poutré, 2002).
- Statistical approach (Pfister, Barber, & Gerstner, 2003).
- Linear algebra formalisms (Carnell & Richardson, 2005).
- Evolutionary Strategy (Belatreche, Maguire, McGinnity, & Wu, 2003).
- Synfire Chains (Sougné, 2000).
- Supervised Hebbian Learning (Legenstein, Naeger, & Maass, 2005; Ruf & Schmitt, 1997).
- Remote Supervision (Kasiński & Ponulak, 2005).

For a detailed review see Kasiński and Ponulak (2006).

SpikeProp (Bohte et al., 2002) is a gradient descent training algorithm for SNNs that is based on backpropagation. The discontinuous nature of spiking neurons causes problems with gradient descent algorithms, but SpikeProp overcomes this issue by only allowing each neuron to fire once and by training the neurons to fire at a desired time. However, if weight updates leave the neuron in a state such that it will not fire, the algorithm cannot restore the neuron to firing for any new input pattern. Additionally, since each neuron is only allowed to fire once, the algorithm can only be used in a time-to-first-spike coding scheme which means that it cannot learn patterns consisting of multiple spikes.

By employing a probabilistic approach to the Hebbian interaction between pre- and post-synaptic firing, it is possible to produce a likelihood that is a smooth function of its parameters (Pfister et al., 2003). The aim of this, of course, is that this allows gradient descent to be applied to the changing of synaptic efficacies. This statistical approach employs STDP-like learning windows and an injected teacher current. Consequently, the method has been described (Kasiński & Ponulak, 2006) as a probabilistic version of Supervised Hebbian learning (Legenstein et al., 2005; Ruf & Schmitt, 1997). Experiments with this approach have been limited to networks consisting of only two spikes, so it is difficult to know how robust the technique would be for larger networks.

Linear algebra formulisms involving definitions of inner product, orthogonality and projection operations for spike time series form the backbone of Carnell and Richardson's work (Carnell & Richardson, 2005). The Gram–Schmitt process is used to find an orthogonal basis for the input time series subspace, and this is then used to find the subspace for the desired output. A batch style iterative process is described that seeks to then minimise the error between target and actual outputs by projecting the error into the input subspace. The Liquid State Machine (LSM) (Maass, Nätschlager, & Markram, 2002) is used for the experiments. Successful training is dependent on the variability of input spikes, but since the training requires batch learning the method is unsuitable for online learning (Carnell & Richardson, 2005).

Evolutionary strategies (ES) have been applied as a form of supervision for SNNs (Belatreche et al., 2003). ES differ from genetic algorithms in that they rely solely on the mutation operator. The accuracy of the resulting SNN provides the basis for determining the fitness function and the ES population was shown to produce convergence to an optimal solution. The learning capabilities of the ES were tested with the XOR and Iris benchmark classification problems. The Spike Response Model was used to model the spiking neurons in a fully connected feed-forward topology. A limitation of this approach is that only the time-to-first-spike is considered by the ES (Belatreche et al., 2003). Additionally, as with all evolutionary algorithms the evolutionary process is time-consuming and this renders them unsuitable for online learning.

A synfire chain (SFC) (Sougné, 2000) is a feed-forward multi-layered topology (chain) in which each pool of neurons in the