

Feature selection to enhance a two-stage evolutionary algorithm in product unit neural networks for complex classification problems

Antonio J. Tallón-Ballesteros^{a,*}, César Hervás-Martínez^b, José C. Riquelme^a, Roberto Ruiz^c

^a Department of Languages and Computer Systems, University of Seville, Reina Mercedes Avenue, Seville 41012, Spain

^b Department of Computer Science and Numerical Analysis, University of Córdoba, Campus of Rabanales, Albert Einstein Building, Córdoba 14071, Spain

^c Area of Computer Science, Pablo de Olavide University, Km. 1, Utrera Road, Seville 41013, Spain

ARTICLE INFO

Available online 24 October 2012

Keywords:

Artificial neural networks
Product units
Evolutionary algorithms
Classification
Feature selection
High error problems

ABSTRACT

This paper combines feature selection methods with a two-stage evolutionary classifier based on product unit neural networks. The enhanced methodology has been tried out with four filters using 18 data sets that report test error rates about 20 % or above with reference classifiers such as C4.5 or 1-NN. The proposal has also been evaluated in a liver-transplantation real-world problem with serious troubles in the data distribution and classifiers get low performance. The study includes an overall empirical comparison between the models obtained with and without feature selection using different kind of neural networks, like RBF, MLP and other state-of-the-art classifiers. Statistical tests show that our proposal significantly improves the test accuracy of the previous models. The reduction percentage in the number of inputs is, on average, above 55 %, thus a greater efficiency is achieved.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

There are several machine learning techniques to deal with a classification problem, such as neural networks, radial basis functions, rules and decision trees. A review of them can be found in [1]. The explosion of available information complicates this problem. Moreover, redundancy or noise may be present on data [2]. Neural networks models play a crucial role in pattern recognition [3]. For many practical problems, the possible inputs to an Artificial Neural Network (ANN) can be huge. There may be some redundancy among different inputs. A large number of inputs to an ANN increase its size and thus require more training data and longer training times in order to achieve reasonable generalization ability. Pre-processing is often needed to reduce the number of inputs to an ANN. The application of feature selection (FS) approaches has become a real prerequisite for model building due to the multi-dimensional nature of many modelling task in some fields. Theoretically, having more features should give us more discriminating power. However, this can cause several problems: an increased computational complexity and cost, too many redundant or irrelevant features, and degradation in the classification error estimation.

Our objective is to improve the accuracy and to reduce the complexity (measured by means of the number of inputs) of the

models of Evolutionary ANNs (EANNs) with product units (PUs) that have been employed to date by us. The training of databases for classification, which have different numbers of patterns, features and classes, is dealt with by means of ANNs. The computational cost is very high if Evolutionary Algorithms (EAs) with different parameter settings are employed for the training of the above-mentioned networks. However, in this paper we use a specialization of an EA called TSEA (Two-Stage Evolutionary Algorithm) [4] which add broader diversity at the beginning of the evolution. First of all, FS is applied to the data sets in order to eliminate redundant and irrelevant variables. In this way, the complexity could be reduced and the accuracy could be increased. The reduction in the number of inputs could decrement the number of nodes in the hidden-layer and, hence, also simplify the associated model. Several runs of the TSEA have been performed to smooth the stochastic character using mean values in order to complete a statistical analysis of the results obtained. This paper is organized as follows: Section 2 describes some concepts about FS and the classification with TSEA in evolutionary product unit neural networks (PUNNs); Section 3 presents the description of our proposal; Section 4 details the experimentation process; then Section 5 shows and analyzes the results obtained; finally, Section 6 states the concluding remarks.

2. Methodology

2.1. Feature selection

The selection of features and the removal or reduction of redundant information unrelated to the classification task on

* Corresponding author. Tel.: +34 954556237; fax: +34 954557139.

E-mail addresses: atallon@us.es (A.J. Tallón-Ballesteros), chervas@uco.es (C. Hervás-Martínez), riquelme@us.es (J.C. Riquelme), robertoruiz@upo.es (R. Ruiz).

hand will not only reduce the complexity of the problem and improve the efficiency of the processing but also simplify significantly the design of the classifier. The FS is one of the essential and frequently used techniques in machine learning. A FS method generates different candidates from the feature space and assesses them based on an evaluation criterion to find the best feature subset [5]. On the basis of the evaluation criterion, FS can be divided into filter methods and wrapper methods. Filters assess the relevance of features by looking only at the intrinsic properties of the data, such as distance, consistency, and correlation [5–7]. These criteria are independent of any inductive learning algorithm. In contrast, the wrapper approach requires one predetermined mining algorithm and uses its performance to evaluate and determine which features are selected [8]. Wrappers often select features that have a higher accuracy; however, they are criticized for their high computational cost and low generality. To take advantage of the above two approaches, a hybrid model was proposed to handle large data sets [9]. Moreover, some methods, known as embedded, use internal information of the classification model to perform FS [10,11].

Based on the generation procedure, FS can be divided into individual feature ranking (FR) and feature subset selection (FSS) [10,12]. FR measures the relevance of each feature to the class and then ranks features by their scores and selects the top-ranked features. These methods are widely used because of their simplicity, scalability, and good empirical success [10,13]. However, FR is criticized because it can capture only the relevance of the features to the target concept, whereas the redundancy and basic interactions between features are not discovered. Additionally, the number of features retained is difficult to determine; as a result, a threshold is required. In contrast, FSS attempts to find a set of features that have good performance. This method integrates the metric for measuring the feature–class relevance and the feature–feature interactions. In [14] Liu and Yu, a large number of selection methods are categorized, in which different algorithms address these issues distinctively. We found different search strategies, namely exhaustive, heuristic and random searches, and combined them with several types of measures to form different algorithms. The time complexity is exponential in terms of the data dimensionality for an exhaustive search, and it is quadratic for a heuristic search. The complexity can be linear with the number of iterations in a random search, but experiments show that, to find the best feature subset, the number of iterations required is usually at least quadratic to the number of features [15]. In this categorization, to handle large data sets, a hybrid model was also proposed to combine the advantages of the FR and FSS techniques. These methods decouple relevance analysis and redundancy analysis, and they have been proven to be more effective than ranking methods and more efficient than subset evaluation methods on many traditional high-dimensional data sets. In this framework, [16] proposed a hybrid search algorithm. Yu and Liu [17] proposed a fast correlation-based filter algorithm (FCBF) that used a correlation measure to obtain relevant features and to remove redundancy. Ding and Peng [18] used mutual information for gene selection, finding maximum relevance with minimal redundancy by solving a simple two-objective optimization.

2.2. Classification with evolutionary product unit neural networks based on a two-stage algorithm

There are several kinds of neural networks, being the single-hidden-layer feed-forward network architecture the most popular one. Multiplicative neural networks contain nodes that multiply their inputs instead of adding them. This class of neural networks comprises such types as sigma-pi networks and product

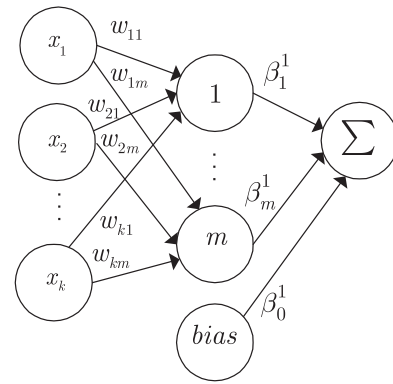


Fig. 1. Structure of a product unit neural network model for a bi-classification problem.

unit networks. The latter type was introduced by R. Durbin and D. Rumelhart [19]. The methodology employed here consists of the use of an EA as a tool for learning the architecture and weights of a PUNN model [20]. More details about PUNNs, such as some of the advantages, the universal approximation theorem, problems and learning methods, can be found in [4,21].

Fig. 1 shows the structure of a PUNN model with a $k:m:1$ architecture for a bi-classification problem; this is a three-layer architecture, that is, k nodes in the input layer, m ones (product units) and a bias one in the hidden layer and one node in the output layer.

The transfer function of each node in the hidden and output layers is the identity function. Thus, the functional model obtained by each of the nodes in the output layer with J classes is given by:

$$f(x_1, x_2, \dots, x_k) = \beta_0^l + \sum_{j=1}^m \beta_j^l \left(\prod_{i=1}^k x_i^{w_{ij}} \right) \quad l = 1, 2, \dots, J; w_{ij} \in \mathfrak{R} \quad (1)$$

Next, we are going to describe briefly the TSEA applied. A full explanation of it and the details about common parameters can be read in Section 3 of [4]. TSEA is used to design the structure and learn the weights of PUNNs in two sequential phases. The population is subjected to the operations of replication and mutation; two types of mutations have been applied: parametric and structural ones. The TSEA pseudo-code for a classification problem appears in Fig. 2. In the first stage, TSEA evolves two populations for a small number of generations. The best half individuals of each one are merged in a new population that follows the full evolutionary cycle. The main parameters of the TSEA are the maximum number of generations (gen) and the maximum number of nodes in the hidden layer (neu). The minimum number of nodes is an unit lower than neu. The remaining parameters will be described further on. At the end of the TSEA, it returns the best PUNN model with a number of nodes between neu and $neu + 1$ in the hidden layer.

We have considered a standard soft-max activation function, associated with the g network model, given by:

$$g_j(\mathbf{x}) = \frac{\exp f_j(\mathbf{x})}{\sum_{j=1}^J \exp f_j(\mathbf{x})} \quad j = 1, \dots, J \quad (2)$$

where J is the number of classes in the problem, $f_j(\mathbf{x})$ is the output of node j for pattern \mathbf{x} and $g_j(\mathbf{x})$ is the probability that this pattern belongs to class j .

Given a training set $D = (\mathbf{x}_i, \mathbf{y}_i) i = 1, \dots, N$, a function of cross-entropy error is used to evaluate a network g with the instances of a problem

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات