Biologically Inspired Cognitive Architectures (2014) 9, 57–70

Available at www.sciencedirect.com

ScienceDirect

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

RESEARCH ARTICLE

Hybrid expert system using case based reasoning and neural network for classification

Saroj Kr. Biswas , Nidul Sinha, Biswajit Purakayastha, Leniency Marbaniang

NIT, Silchar 788010, Assam, India

Received 5 March 2014; received in revised form 19 May 2014; accepted 7 June 2014

KEYWORDS
Case based reasoning;
Artificial neural networks;
Similarity measure;
k-NN similarity measure;
Artificial intelligence;
Data mining

Abstract
Case Based Reasoning (CBR) is an analogical reasoning method, which solves problems by relating some previously solved problems to a current unsolved problem to draw analogical inferences for problem solving. But CBR faces the challenge of assigning weights to the features to measure similarity between a current unsolved case and cases stored in the case base effectively and correctly. The concept of neural network's pruning is already used to sort out feature weighting problem in CBR. But it loses generality and actual elicited knowledge in the ANN's links. This work proposes a method to extract symbolic weights from a trained neural network by observing the whole trained neural network as an AND/OR graph and then finds solution for each node that becomes the weight of a corresponding node. The proposed feature weighting mechanism is used in CBR to build a hybrid expert system for classification task and the performance of the proposed hybrid system is compared with that with other feature weighting mechanisms. The performance is validated on swine flu dataset and ionosphere, sonar and heart datasets collected from UCI repository. From the empirical results it is observed that in all the experiments the proposed feature weighting mechanism outperforms most of the earlier weighting mechanisms extracted from trained neural network.

Introduction

One of the objectives of computational intelligence is to impart the systems with the ability to reproduce human like reasoning. Case Based Reasoning (CBR) is a variety of

Corresponding author. Tel.: +91 09508727987.
E-mail address: bissarojkum@yahoo.com (S.Kr. Biswas).

http://dx.doi.org/10.1016/j.bica.2014.06.004
2212-683X/

reasoning by analogy (Aamodt & Plaza, 1994; Leake, 1996). It is an artificial intelligence approach to learning and problem solving based on past experiences stored in a case base and it also captures new knowledge/experiences, making it immediately available for solving next problems. These experiences encode relevant features/attributes, courses of action that were taken, and solutions that ensued. This base of experience forms the memory for the CBR system. Aamodt and Plaza (1994) have described CBR typically as a cyclical process comprising the four REs:

- *Retrieval* which retrieves one or more similar cases from the case base that can be used to solve a target problem. It starts with a partial problem's description and ends when finds the most similar previous case/cases.
- *Reuse* is responsible for proposing solution to the target problem from retrieved cases.
- *Revise* is responsible to evaluate the retrieved solution. If retrieved solution is fit for the target case it is then possible to learn about the success, otherwise the solution is repaired/adapted using some problem domain's specific knowledge or any other ways.
- *Retain* consists of a process of integrating the useful information about the target case's resolution in the case-base.

The core of CBR methodology is the retrieval of cases stored in a case base, which are very similar to a query (target) case and thus a similarity measure is required to calculate the similarity between stored cases and the query case. Hence, similarity measures are the key elements in obtaining a reliable solution (classification) for new situations (Buta, 1994; Nunez et al., 2004). The task of defining similarity measures for real world problems is one of the greatest challenges of research in this area as assessing the similarity between cases is a key aspect of the retrieval phase in CBR. The most popular similarity measure is k nearest neighbor (k-NN), which uses a distance function to generate predictions from stored cases. The biggest problem here is to determine the weight of the features as several studies have shown that k-NN's performance is highly sensitive to the definition of its distance function (Watson & Marir, 1994; Wettschereck, Aha, & Mohri, 1997). Many k-NN variants have been proposed to reduce this sensitivity by parameterizing the distance function with feature weights (Wettschereck et al. 1997). k-NN variants are also frequently used for case retrieval in CBR. k-NN considers that each query q is represented by n features which are numeric or discrete. The similarity of q with each stored case is calculated where each case is represented as $x = \{x_1, x_2, x_3, \ldots, x_n, x_c\}$ in a set X, x_1 to x_n are attribute values or problem description of the case x and x_c is x's class value. k-NN then retrieves the k most similar (least distance) cases and predicts their weighted majority class or majority class only as the class of q. The distance can be calculated by Eq. (1) given below.

$$\text{distance}(x, q) = \sqrt{\sum\nolimits_{i=1}^{n} w_i \text{diff}\,(x_i, q_i)^2} \tag{1}$$

where w_i is the parameterized weight value assigned to feature i and

$$\text{diff}\,(x_i, q_i) = \begin{pmatrix} |x_i - q_i| & \text{if feature } i \text{ is numeric} \\ 0 & \text{if feature } i \text{ is discrete and } x_i = q_i \\ 1 & \text{otherwise} \end{pmatrix} \tag{2}$$

The distance given in Eq. (1) is weighted Euclidean distance but it can also be weighted absolute or city block distance. The concept of equal weights handicaps k-NN as it allows redundant and irrelevant features to have as much impact on distance computations as other features. For the cases belonging to the same class, some features may often have the same value, while others vary their values in most of the cases in that class. Therefore, the features will always have different degree of impact in retrieving similar cases from the case base. Accordingly, different feature weights should be provided to avoid incorrectness in classification/prediction task. If all of the features are regarded as being equally important, i.e., all the features have the same weight value, CBR allows redundant or irrelevant features to influence the prediction. So, it is very important to solve the feature weighting problem of CBR to work properly where similarity measure is k-NN. Many methods have been proposed to sort out the feature weighting problem that instead assign higher weight setting to the more relevant features for case retrieval. Although many feature weighting methods for k-NN have been reported for classification/prediction task, feature weighting methods which can capture generality and domain specific knowledge together are rare. For example Daelemans, Gillis, and Durieux (1994) and Wettschereck and Dietterich (1995) used mutual information to compute coefficients on numeric attributes. Many other feature weighting methods and their analysis could be found and are available in a review by Wettschereck et al. (1997).

The mechanism used in this paper is feature weight extraction, which captures generality and domain intensive knowledge to estimate the relative importance of each feature. When properly weighted, an important feature would receive a larger weight than less important or irrelevant features. The feature weighting mechanism of this work is based on a trained neural network. The importance of a feature is mined from the strengths of connected links in a trained neural network. The explanation behind this idea can be given as an important feature should have strong links/connections along the nodes correlated to this feature, because of its influence on classification/prediction task. The advantage of using a neural network for feature weighting is that the artificial neural networks (ANN's) are well-known massively parallel computing models which exhibit excellent behavior in input-output mapping and resolving complex artificial intelligence problems in forecasting and classification tasks.

Some researchers have used ANNs to extract symbolic rules (For example Craven and Shavlik (1997)) and the rules generated by them are in the form of a decision tree and many works have also been done at network pruning (Lu & Setiono, 1996). Many network pruning tasks are also done (Ha 2008; Im & Park, 2007; Park, Im, Shin, & Park, 2004; Park, Kim, & Im, 2006; Peng & Zhuang, 2007; Sarwar, Ul-Qayyum, & Malik, 2010; Shin & Park, 1999; Shin, Yun, Kim, & Park, 2000; Yang & Jin, 2010; Zeng & Martinez, 2004) to find feature weight to sort out feature weighting