

# A joint generalized exemplar method for classification of massive datasets



Mehmet Emin Tağluk<sup>a</sup>, Ömer Faruk Ertuğrul<sup>b,\*</sup>

<sup>a</sup> Electrical and Electronic Engineering, İnönü University, 44060 Malatya, Turkey

<sup>b</sup> Electrical and Electronic Engineering, Batman University, 72060 Batman, Turkey

## ARTICLE INFO

### Article history:

Received 23 February 2015

Received in revised form 18 July 2015

Accepted 30 July 2015

Available online 8 August 2015

### Keywords:

Nested generalized exemplar

Exemplar-based learning

Classification

Compression

Artificial intelligence

## ABSTRACT

Due to technological improvements, the number and volume of datasets are considerably increasing and bring about the need for additional memory and computational complexity. To work with massive datasets in an efficient way; feature selection, data reduction, rule based and exemplar based methods have been introduced. This study presents a method, which may be called joint generalized exemplar (JGE), for classification of massive datasets. This method aims to enhance the computational performance of NGE by working against nesting and overlapping of hyper-rectangles with reassessing the overlapping parts with the same procedure repeatedly and joining non-overlapped hyper-rectangle sections that falling within the same class. This provides an opportunity to have adaptive decision boundaries, and also employing batch data searching instead of incremental searching. Later, the classification was done in accordance with the distance between each particular query and generalized exemplars. The accuracy and time requirements for classification of synthetic datasets and a benchmark dataset obtained by JGE, NGE and other popular machine learning methods were compared and the achieved results by JGE found acceptable.

© 2015 Elsevier B.V. All rights reserved.

## 1. Introduction

Recent technological developments brought about dealing with massive data particularly in communication, computers and storage systems. As a result of the need for dealing with immense datasets, the demand and utilization of machine learning (ML) methods have been increasing in various areas. These methods are employed to model systems, generalize experimental results or make decisions. Although massive datasets increases the accuracy of ML, it also increases computation and storage costs. The increase of the number of instances in the training set naturally leads to computational burden, memory requirements and also sensitivity to noise [1].

In the k nearest neighbor (kNN) [2], and other kNN based methods [3–5], the decision are made by calculating the distance, which is a similarity measure [6], between the query and all the examples in the training set [7,8]. The optimum parameters such as weights and biases for an artificial neural network (ANN) are determined by utilizing whole training dataset, while training ANN by backpropagation [9]. Such processes indeed require a heavy computational

work, particularly when a massive dataset is processed. Therefore, to overcome the computational work without loss of accuracy, novel ML methods are being developed and the training schemes of existing ML methods are being improved continuously. For example, extreme learning machine (ELM) [10], which has been proposed to overcome training time via computing the output weights analytically instead of tuning in backpropagation [11].

For reduction of the size of datasets: (a) feature selection [12–15], (b) data condensing [16–19] and (c) rule based [20–22] methods have been presented in the literature. Feature selection methods are employed to determine the best subset of data by eliminating the irrelevant, redundant or noisy features. The major outcomes of these methods are improvement of the accuracy of ML [12–15], reduction of the computational cost and memory requirements in each particular stage (such as: communication, measurement, storage and processing stages) [13,14]. In data condensing methods the data is reduced by removing the overlapping or unnecessary observations or experiments in the training dataset in order to gain a computational advantage [16–18]. Details of these data reduction methods may be followed in the reviews of Wilson and Martinez [1,23]. In the rule based methods, only the rules, which are provided from the training dataset, are memorized [9,22,24], such as in ANN. After the training stage of ANN, only weights, biases, transfer function and network architecture

\* Corresponding author. Tel.: +90 488 2173500; fax: +90 488 2153601.

E-mail address: [omerfaruk.ertugrul@batman.edu.tr](mailto:omerfaruk.ertugrul@batman.edu.tr) (Ö.F. Ertuğrul).

are stored. A trained ANN only uses these fixed parameters to make a decision on a further query [9].

Another marginal solution for reducing dataset without degrading accuracy is using an exemplar-based learning method [1,25], which is based on inductive learning method [26–28]. In the exemplar-based learning methods, the general descriptions of examples that form concepts such as hypothesis or hyper-rectangles (HR), so called generalized exemplars (GE), are used instead of using several training examples in the training dataset [25,27,29]. Each of these solutions are humanoid approaches since human learn by reducing the complex environment and various stimuli with: (a) categorization by grouping the objects or stimuli that have some common physical or functional traits; (b) defining an ideal exemplar as a prototype that sums up the characteristics of members in the same category [19,30,31]; and (c) extracting rules to better control or understands [1,26,32,33]. Some researchers reported that the exemplars are stored in human memory, to form category definitions, but in the instant-based ML methods, such as kNN, all training examples are retained in memory and never been changed [19,34,35]. All of the studies reviewed so far, showed that employing GEs in place of the whole dataset not only reduce the storage and computational costs [36], but also improve generalization accuracy by avoiding noise and overfitting [1].

In 1991, Salzberg had published a paper describing nested GE (NGE) theory [25], in which NGE has been simply defined as an application of kNN to the axis-parallel HRs (instead of the whole dataset) in  $n$ -dimensional Euclidean space. In NGE, a GE, which may be in the form of an HR or a point (that is known as trivial HR [27,36]), is learned for each class. The query is simply classified by computing the distance (weighted Euclidean distance) among the query and each HRs in the feature space. The class of the query is assigned by the class of its closest HR, but if the query is inside a GE its distance is zero. If there is more than one HRs (e.g. the query may be in the intersection of two HRs) near the query, the class of the query is determined by the class of HR, which has smallest volume [25,29,37]. The weights, which are utilized in distance calculations, are determined by the accuracy of predictions in the learning stage. The correct predictions strengthen the weight and the size of HR [25], while the wrong ones weaken these weights. The practice of this classification process has discussed in detail by Aha [38]. NGE was compared with other incremental learning methods such as fuzzy ARTMAP, generalized fuzzy min-max neural networks, growing neural gas and incremental learning, which based on function decomposition methods [39]. The obtained results had shown that NGE can be successfully used for classification. Wettschereck had reported advantages of NGE in data compression, fast learning and classification [29]. On the other hand, there are some difficulties with NGE like nesting HRs [26,27,29,37], overlapping HRs [26,27,29,37], search algorithm [37], the decision boundaries (shape of GE is HR) [37], search direction [37] and the seed order of data [40]. The main properties of improved versions of NGE, which are greedy NGE [25], NONGE, OBNGE, BNGE, BNGE FW<sub>MI</sub> [37], KBNGE [29], NNGE [26] and iDTt-NGE [41], are summarized in Table 1.

The fundamental idea behind the method proposed in this study is to show that each member in the dataset can be separated into GEs without allowing any nesting or overlapping. For this propose, an iterative process was designed. In this process, an HR is fitted or determined for each class. If the HRs are intersects, then the non-overlapping parts of HRs are labeled in according to the underlying data class, the overlapping part/parts are isolated and put through the same process with smaller hyper rectangles. This process is followed on until all of data are separated according to their characteristic classes. Then the labeled HR sections (small and big) are joined together according to their label. In this way, the joint generalized exemplars are obtained in the feature space.

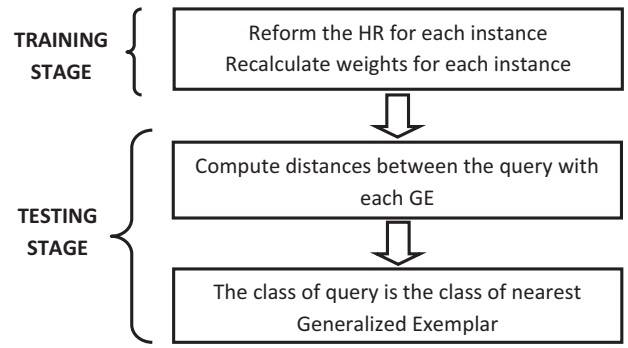


Fig. 1. NGE algorithm.

To evaluate and validate this approach, the JGE is tested for synthetic datasets that have different characteristics and a benchmark dataset. The results were compared with the results obtained through popular classification methods (Nearest Mean Classifier (NMC), k Nearest Neighbor (kNN), Naïve Bayes (NB), Support Vector Machines (SVM), ANN, Decision Trees (DT)) for assessing the performance of the new method in terms of accuracy and computational speed. The proposed method was found successful enough to be applied to any classification problem.

The rest of the paper is organized as follows. Section 2 explains a brief overview of NGE, the mathematical background of JGE and improvements of JGE. Section 3 presents some examples to show the efficiency of the proposed method. Section 4 concludes the performed paper.

## 2. Method

### 2.1. NGE

NGE learns by reforming HRs over training dataset for each particular data incrementally either expanding or shrinking way, allowing both nesting and overlapping [39]. NGE can be simplified as shown in Fig. 1. In Fig. 2(a and b), two different examples are given to show HR sets for different types of datasets. The data in Fig. 2(a) can easily be classified, but the one given in Fig. 2(b) is a little bit complex data and it cannot easily be distinguished [25,37].

In the training stage, an HR is fitted to each data class. The class of a query is assigned by calculating the distance between the query and each of the HRs [25] by weighted Euclidean distance method as showed below.

$$D_{EH} = w_{H^j} \sqrt{\sum_{i=1}^n \left( w_{f_i} \frac{d_{f_i}}{\max_i - \min_i} \right)^2} \quad (1)$$

$$d_{f_i} = \begin{cases} E_{f_i} - H_{upper}^j & \text{when } E_{f_i} > H_{upper}^j \\ H_{lower}^j - E_{f_i} & \text{when } E_{f_i} < H_{lower}^j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where  $E$  refers to query,  $H$  hyper-rectangle,  $E_{f_i}$  is the value of the  $i$ th feature of  $E$ ,  $\max_i$  and  $\min_i$  are the maximum and minimum values of  $i$ th feature,  $n$  is the number of features,  $H_{lower}^j$  is the lower left corner and  $H_{upper}^j$  is the upper right corner [25,37]. The weight of  $j$ th HR ( $w_{H^j}$ ), which is applied for increasing the tolerance of NGE to noise [25], is calculated as follows [37]:

$$w_{H^j} = \frac{\text{Number of comparison of } H^j \text{ with } E_x}{\text{Number of } (H^j \text{ equals to } E_x)} \quad (3)$$

$w_{H^j}$  was also defined as the inverse measure of reliability that shows the inverse of how frequently the prediction is correct for

Download English Version:

<https://daneshyari.com/en/article/494829>

Download Persian Version:

<https://daneshyari.com/article/494829>

[Daneshyari.com](https://daneshyari.com)