Engineering Applications of Artificial Intelligence 37 (2015) 392–406

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

An on-line weighted ensemble of regressor models to handle concept drifts

Symone Gomes Soares*, Rui Araújo

Institute for Systems and Robotics, and Department of Electrical and Computer Engineering,University of Coimbra, Pólo II, PT-3030-290 - Coimbra, Portugal

ARTICLE INFO

Article history:
Received 25 August 2014
Received in revised form
30 September 2014
Accepted 4 October 2014
Available online 1 November 2014

Keywords:
Concept drift
Ensemble learning
Learning in changing environments
Regression
Ensemble pruning strategies

ABSTRACT

Many estimation, prediction, and learning applications have a dynamic nature. One of the most important challenges in machine learning is dealing with concept changes. Underlying changes may make the model designed on old data, inconsistent with new data. Also, algorithms usually specialize in one type of change. Other challenge is reusing previously acquired information in scenarios where changes may recur. This strategy improves the learning accuracy and reduces the processing time. Unfortunately, most existing learning algorithms to deal with changes are adapted on a batch basis. This process usually requires a long time, and such data may not reflect the current state of the system. However, even the system is adapted on a sample basis, existing algorithms may adapt slowly to changes and cannot conciliate old and new information. This paper proposes an On-line Weighted Ensemble (OWE) of regressor models which is able to learn incrementally sample by sample in the presence of several types of changes and simultaneously retain old information in recurring scenarios. The key idea is to keep a moving window that slides when a new sample is available. The error of each model on the current window is determined using a boosting strategy that assigns small errors to the models that predict accurately the samples predicted poorly by the ensemble. To handle recurring and non-recurring changes, OWE uses a new assignment of models' weights that takes into account the models' errors on the past and current windows using a discounting factor that decreases or increases the contribution of old windows. In addition, OWE launches new models if the system's accuracy is decreasing, and it can exclude inaccurate models over time. Experiments with artificial and industrial data reveal that in most cases OWE outperforms other state-of-the-art concept drift approaches.

1. Introduction

On-line learning applications where the target concept may change over time pose serious problems. Underlying changes may make the model designed on old data, inconsistent with new data. This problem is known as *concept drift*. One of the most important challenges in machine learning is dealing with concept changes. Other challenge in on-line learning algorithms is to adapt and deal with changes without being informed about them, and make use of the past experiences in situations where old contexts may reappear (Widmer and Kubat, 1996; Bosnić et al., 2014). In industry, a large demand of algorithms for on-line prediction is observed, such algorithms are usually called Soft Sensors (Ibargüengoytia et al., 2013; Kadlec et al., 2011). They employ predictive models to provide on-line estimations of difficult-to-measure variables based on some easy-to-measure variables (Wu et al., 2009). Examples of applications are the groundwater level prediction in a coastal aquifer (Taormina et al., 2012), the river flow discharge prediction in a reservoir (Cheng et al., 2005), and others (Kadlec and Gabrys, 2011; Chau, 2007). Unfortunately, industrial processes exhibit time-varying behavior. Causes for such a behavior are changes in the measuring devices, environment changes, and changes of process behavior or of some external process condition (Vergara et al., 2012). It is important to develop on-line adaptive methodologies that should be able to handle time varying behavior in prediction settings.

The main adaptive mechanisms to deal with concept drift can be classified as: *instance selection* ($A1$), *instance weighting* ($A2$), and *ensemble learning* ($A3$). Instance selection approaches obtain a set of relevant samples of the current concept. A common strategy is the moving window (MW), in which a window slides along the data and the newest samples are included to be taken into consideration into the model and the oldest samples are excluded from the model. One difficulty is the selection of the window's size. In instance weighting (IW), samples are weighted according to their age and/or relevance to the current concept. Recursive methods, where the models' parameters are updated over time, belong to this category. The method most commonly used is the Recursive Partial Least Squares

* Corresponding author. Tel.: +351 914158717.
E-mail addresses: symonesoares@isr.uc.pt (S. Gomes Soares), rui@isr.uc.pt (R. Araújo).

http://dx.doi.org/10.1016/j.engappai.2014.10.003
0952-1976/

(RPLS) proposed in Qin (1998). In this work, a currently existing Partial Least Squares (PLS) model is merged with the new data using a forgetting factor that is employed to update the model. Recent RPLS models update the mean and variance of data using a MW approach to track the process changes (Ahmed et al., 2009). In general, the RPLS works well in cases where the process dynamics are well represented in the initial training data set. On the contrary, the model may not track the process dynamics occurring in the new data. Boosting is also an IW approach, where samples are re-weighted in order to emphasize those samples predicted incorrectly by the previous model (Drucker, 1997).

Ensemble learning has been proven itself as a valuable tool to handle concept drift scenarios. It combines a set of models in order to get a final prediction (Chao et al., 2014; Huang and Chau, 2008). Results indicate that ensemble learning improves the generalization capability and the overall system performance (Soares et al., 2012; D'Este et al., 2014). Adaptive ensembles can combine a subset of the following strategies (Polikar, 2012): adaptation of the models' weights ($E1$); adaptation of the models' parameters ($E3$); and/or add new models in the ensemble ($E3$) according to each incoming sample or batch of samples. Adaptive ensembles can also be classified as *batch-based* or *sample-based* if they are adapted when a batch of data or a sample is available, respectively.

Most adaptive ensembles are batch-based and focus on classification tasks, for example the Learn^{++}.NSE (Elwell and Polikar, 2011), which is inspired by the Boosting (Drucker, 1997). When a batch arrives, Learn^{++}.NSE identifies the samples misclassified by the ensemble and obtains a penalty distribution. The distribution is used to assign errors to each model based on its contribution to the ensemble. The weight of each model is assigned using a weighted average of their errors on the current and old batches by a sigmoid function with two slope parameters. The parameters' setting is not an easy task, since Learn^{++}.NSE is sensitive to their values. Moreover, Learn^{++}.NSE requires a long time for waiting a batch, and when it is available, it may not reflect the current concept.

On the contrary, sample-based ensembles offer faster adaptation capability. Examples are the Additive Expert (AddExp) (Kolter and Maloof, 2005) and the Incremental Local Learning Soft Sensing Algorithm (ILLSA) (Kadlec and Gabrys, 2011) algorithms. AddExp uses a loss bound to obtain the models' errors, and models' weights are adapted according to their actual losses and a decreasing factor, employed to reduce a model's weight when it performs poorly. The ILLSA algorithm has two phases. On the training phase, a set of models is designed, where each model is trained with samples of a different concept contained on the training data; while on the on-line phase, for each incoming sample, the models' weights are adapted using the posterior probability by a Bayesian framework. ILLSA works well when the process dynamics are well represented in the training data set. One drawback is that few models are designed if the training data contains few concepts.

This paper proposes an on-line weighted ensemble of regressor models (OWE) which is able to learn incrementally sample by sample in the presence of several types of changes and simultaneously retain old information in recurring scenarios. OWE employs several adaptive mechanisms to deal with several types of drifts. OWE is inspired by Learn^{++}.NSE (Elwell and Polikar, 2011). But unlike Learn^{++}.NSE, in the OWE, the ensemble is adapted on a sample basis, leading the system to faster recovery from changes and increasing the system accuracy. Additional and new strategies are proposed to increase the OWE's accuracy. The experiments indicate that OWE outperforms Learn^{++}.NSE in all tests.

The key idea is to keep a fixed MW slides along data when a new sample is available. Then, the error of each model on the current window is determined using a boosting strategy (Feely, 2000; Shrestha and Solomatine, 2006) that assigns small errors to the models that predict accurately the samples predicted poorly by the ensemble. To handle recurring and non-recurring changes, OWE uses a new method for assigning the models' weights that takes into account the models' errors on the past and recent windows using a discounting factor that decreases or increases the contribution of old windows. In addition, OWE launches new models if the system's accuracy is decreasing, and it can remove inaccurate models for reducing memory and computational time. The method removes the model with the largest total error rate on the current and old windows. Experiments on artificial data sets and industrial data sets are detailed to evaluate, and demonstrate the performance and the effectiveness of the OWE over the state-of-the-art concept drift approaches.

The main contributions of this work are (1) a new on-line weighted ensemble of regressor models with faster adaptation capability; (2) regression scope (while most on-line ensemble applications for handling changes is devoted to classification tasks); (3) systematic analysis of the related ensemble algorithms; (4) thorough analysis of the experimental results using both artificial data sets and industrial data sets, demonstrating faster adaptation capability and accuracy of the OWE over the main state-of-the-art approaches; (5) implementation of a new Learn^{++}.NSE algorithm for regression tasks.

The paper is organized as follows. Section 2 presents the theory of concept drift. Section 3 outlines the related works. In Section 4 the OWE algorithm is described. In Section 5 the results are presented and discussed. Finally, Section 6 presents the concluding remarks.

2. The concept drift problem

For analyzing the problem of concept drift (Kadlec and Gabrys, 2011; Klinkenberg, 2005), consider an on-line learning framework where samples arrive incrementally one by one, and each sample $\mathbf{s} = (\mathbf{x}, y)$ is composed of r inputs grouped into an input vector, $\mathbf{x} \in \mathbb{R}^{r \times 1}$, and one output $y \in \mathbb{R}$. Consider that the samples are grouped into several windows of equal size m:

$$\underbrace{\mathbf{s}_{(1,1)}, \ldots, \mathbf{s}_{(1,m)}}_{\text{window } 1}, \underbrace{\mathbf{s}_{(2,1)}, \ldots, \mathbf{s}_{(2,m)}}_{\text{window } 2}, \ldots,$$
$$\underbrace{\mathbf{s}_{(t,1)}, \ldots, \mathbf{s}_{(t,m)}}_{\text{window } t}, \underbrace{\mathbf{s}_{(t+1,1)}, \ldots, \mathbf{s}_{(t+1,m)}}_{\text{window } t+1},$$

where $\mathbf{s}_{(w,i)}$ is the i-th sample of window w. For each window w, the data is independently and identically distributed according to a distribution $\mathcal{D}_w(\mathbf{x}, y)$. If all the windows are distributed over the same distribution, the concept is considered stable and thus there is no concept drift. On the other hand, if two windows a and b have different data distributions, i.e. $\mathcal{D}_a(\mathbf{x}, y) \neq \mathcal{D}_b(\mathbf{x}, y)$, there is a concept drift. Learning algorithms to handle the concept drift problem should be able to predict the next data window (e.g. $t+1$) using the old data windows (from 1 to t) or a subset of them.

Changes over time may occur in different forms. In the literature, drifts are classified with respect to their speed, cyclical nature, scope, etc (Minku et al., 2010; Elwell and Polikar, 2011; Zliobaite, 2009). The drift speed describes the rate by which old concepts are substituted by new concepts. An *abrupt drift* happens when an old concept is abruptly replaced by a new concept; while a *gradual drift* happens when an old concept is slowly substituted by a new concept. Gradual drifts are harder to identify since they result in small data shift and lower error prediction when compared to abrupt drifts.

Drifts can also be classified according to their cyclical nature. A *recurring drift* happens if a previously occurring concept recurs after some time; while a *non-recurring drift* happens if a previously occurring concept cannot recur over time. Recurring drifts may occur due to the cyclic nature of a system (e.g. due to the seasons of the year). Other drift classification is with respect to scope. A *local drift* affects only some regions of the instance space; while a *global drift* affects the whole instance space. In local drifts, changes depend on the location in the instance space; And therefore, a