Evaluation of different incremental learning methods for video surveillance scenarios

Samaneh Khoushrou[1]
skhoshrou@inescporto.pt
Jaime S. Cardoso[1]
jaime.cardoso@inescporto.pt
Luís F. Teixeira[2]
luisft@fe.up.pt

[1] INESC TEC (formerly INESC Porto)
[2] Faculdade de Engenharia
Universidade do Porto

Abstract

Motivated by the problem of segmenting and tracking objects in video streams, we investigate the application of incremental learning algorithms for evolving data to the development of robust and flexible video object tracking systems. Most existing learning, which are variations on the static learning schemes, can not cope with many real-life challenges, rendering the technology impractical. We propose to develop models that explicitly learn the number and identity of objects depicted in a given video stream. In this work, we investigate the application of three incremental algorithms for adaptive learning from evolving data streams.

1 Introduction

Much of the recent history of computer vision and visual data understanding in general has focused on very simplified settings in stationary environments, such as, fixed and known number of categories to be recognized or learnt, enough computational resources are available [1, 2]. Yet, if the ultimate goal of computational intelligence is to learn from large volumes of data that come from real applications, then the need for a general framework to learn from and adapt to a non-stationary environment can hardly be overstated.

We will focus on adaptive learning algorithms for evolving data applied to the development of robust and flexible video object tracking systems. In particular, we will tackle the problem of tracking objects in multiple video streams. Most existing models, which are variations on the static learning schemes, cannot cope with many real-life challenges, rendering the technology impractical. The main objective of our work is to contribute with a system capable of continuously extracting new information from streams of visual data, in non-stationary environments. Thus, there is the need for a general framework for learning from and adapting to a non stationary environment. Given new data, such a framework would allow us to learn any novel content, reinforce existing knowledge that is still relevant, and forget what may no longer be relevant, only to be able to recall, if and when such information becomes relevant again in the future [3].

An ideal algorithm possesses incremental learning capabilities in a wild environment , if it meets the following criteria: a) Learn incrementally, b) Handle the concept drift, c) Accommodate new classes in Non-stationary environment, d) Bounded complexity and e) Ability to work with unlabelled data or partially labelled data and finally, to be more specific in video applications: f) Handle Multi-Dimensional(MD) data. In a non-stationary environment, where the number and identity of objects change, we need to deal with unlabelled data which makes use of clustering approaches at least in primary steps of process inevitable.

As material for our approach is the main trajectories of people in the scene, we are interested in clustering parallel data streams. This research field has been divided into two groups:*clustering by example*, and *clustering by variable*. In *clustering by example* data points from the same stream may have been assigned to different clusters. On the other hand, *clustering by variable* treats a stream as one unit and a stream with all its points is assigned to one cluster [4]. So, *clustering by variable* seems to be more efficient in terms of runtime and complexity. In such scenarios, we also may have access to partially labelled data or we may be able to define some classes after a while. To cover these situations, we also need an incremental classification approach which possesses all mentioned characteristics.

To the best of our knowledge, none of the existing clustering and classification approaches fulfil all our demands, however methods like ODAC [5], Correl-Clus [6], and Learn++.NC [3] can suit most of our system demands.

The paper is organised as follows. In the next section, two clustering approaches are reviewed and discussed. An incremental classification method is described in Section 3. A comparative study is presented in section 4. Some primary results are presented in section 5. Section 6 will be devoted to future work. Finally, in Section 7 the paper is drawn to conclusion.

2 Clustering algorithms

Clustering refers to the process of grouping a collection of objects into groups or "clusters" such that objects within the same class are similar in a certain sense, and objects from different classes are dissimilar. Herein, two variable-based data stream clustering approaches are reviewed and their characteristics are discussed.

ODAC [5] is an example of variable based approaches, using tree models for on-line clustering. In this method each observation is processed only once and the system is incrementally updated. For updating, the system computes the dissimilarities between time series and considers the cluster's diameter as the highest dissimilarity between two time series belonging to the same cluster. When a cluster's diameter exceeds a threshold, the cluster is split. One important feature of this system is that the dissimilarity matrix is updated only for the cluster currently under test. So, the time complexity of iterations are constant given the number of examples. It may also decrease with every split occurrence. Moreover, the framework does not need a predefined number of target clusters. Furthermore, it provides a good performance on finding the correct number of clusters. However, there is no statistical confidence on the decision of assignment when the structure expands, which might split the variables. Besides, it can not handle dynamic number and length of high-dimensional data streams, which is more than likely situation in a non-stationary scenario.

In [6] a correlation-based framework, called *Correl-Clus*, is proposed. Unlike the widely-used Euclidean distance which only measures the discrepancy of the data values, this correlation-based distance considers two data streams with similar trends to be close to each other. Correl-Clus continuously receives new data records from all the data streams at each time step. It keeps outputting the clusters for the most recent data streams of fixed length L. For every l time steps, it first computes a compressed representation of the data streams for the time segment [t-l+1, t], discards the raw data, and updates the compressed representation of the data streams for the target clustering time [t-L+1, t]. It then calls a correlation-based k-means algorithm to compute the clustering results. Since the number of clusters k may be changing, Correl-Clus also employs a new algorithm to dynamically adjust k in order to recognize the evolving behaviours of the data streams. This method is proposed to cluster one-dimensional data (1D), where the number and length of streams are fixed.

3 Classification algorithms

Classification is the association of patterns with same set of properties into classes. For on-line applications, system needs to deal with large number and various classes. So, using multiple classifiers could lead to better performance than could be obtained from a single one. *Learn++.NSE (non stationary environment)*, trains a new classifier for each new data it receives and combines these classifiers using a dynamically weighted majority voting. This approach allows the algorithm to recognize, and act according to changes in data distributions [3]. Although this method has shown the most promising performance, it has several shortcomings. For instance, if the ensemble is not pruned, the complexity of model will grow

Table 1: Assessment of different learning algorithms

Category	method	Learn incrementally	Concept drift	Accommodate new class	Unlabelled/ labelled	Complexity	Data
Clustering	ODAC, Correl-Clus	√	√	√	*unlabelled*	*Constraint*	1*D*
Classification	Learn++.NC	√	√	√	*labelled*	*Unconstraint*	*MD*

unbounded over time; on the other hand the typical solution of replacing the oldest models with new one potentially discards important information.

4 Comparative study

An objective comparison of the methods, already explained in previous sections is presented in Table1. It can be concluded, that none of the methods has all the characteristics, so we can not apply them directly to our non-stationary incremental scenario.

5 Experimental Results

The evaluation of already mentioned methods are made using synthetic data sets created with specific definitions, described in following section.

5.1 Dataset

The data sets applied to evaluate clustering approaches were created using a time series generator that produces n time series belonging to a predefined number c_k of clusters with a noise constant β. Each cluster c has a pivot time series p, and the remaining time series are created as p $+\lambda$ where

$$\lambda \approx U(-\beta p, \beta p) \quad (1)$$

We created three data sets with ten variables each, observed along 7K examples, with β=0.3 and especially prepared to test different hypothesis: one cluster(1C), two clusters(2C) and three clusters(3C). The structure of 3C was created as $\{\ \{a_1, a_2, a_8\}, \{a_3, a_4, a_9\}, \{a_5, a_6, a_7, a_{10}\}\}$.

To evaluate the performance of learn++.NC algorithm, Gaussian dataset is applied. This dataset features multi class data, each drawn from a Gaussian distributions. Each class experiences gradual but independent drift. To make this experiment more challenging, class addition and removal are added to drift scenario.

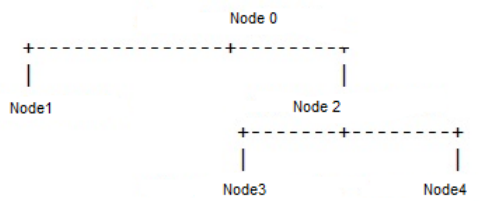

Figure 1: ODAC structure with 3c and 7k observations

5.2 Results

For all these stationary data sets, ODAC finds the correct structure after almost 2k observations, indicating good clustering capabilities. An example of structure is shown in Fig.1.

Correl-Clus clustered fixed size segments (L=200). Right number and structure of clustered data in each batch is considered as accurate clustering otherwise it counted as an error. To compute the accuracy, we normalized the number of correctly clustered batches to total number of batches (total=350). Regarding this definition, accuracy were 80%, 75%, and 56% in 1c, 2c and 3c respectively. In our experiments, ODAC indicated a better and faster clustering capability rather than Correl-Clus.

In our work, a multi-class SVM is applied as base classifier in Learn++.NC algorithm, and the method achieved 57.8% accuracy on Gaussian dataset. It shows a good performance in following drift (updating existing models) and obtaining new models for new observations.

6 Future work

Since we are interested in scenarios where the data is not completely labelled, we will also research the clustering of data streams. These algorithms are fundamental in our setting since, not knowing the classes a priori, one is particularly interested in knowing if a certain object observed at a certain time in a certain camera is the same object being observed in a second camera at a different time. As an object may appear a short period in scene, we will research algorithms working with dynamic number and length of streams. Our methods should also handle multi-dimensional data.

Ensemble-based approaches, herein Learn++.NC, suffer from an uncontrolled growth of complexity. will address this issue by researching new methods to simplify an ensemble of classifiers by a new set with lower cardinality. Intimately related with this issue is the concept of similarity between two different classifiers.

7 Conclusions

A brief overview of three incremental learning algorithms was explained and an objective comparison is presented. Hence, they showed good performance in non-stationary scenarios, but none of these methods fulfil demands for a multi camera video surveillance scenario. ODAC shows better on-line clustering performance than Correl-Clus. To handle labelled data in non stationary environments, Learn++.NSE is applied.

Acknowledgements

This work is financed by the ERDF-European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT-Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP - 01-0124-FEDER-022701.

The authors would like to thank Fundação para a Ciência e a Tecnologia (FCT) for the financial support of PhD grant SFRH/BD/80013/2011.

References

[1] L. Fu, "Incremental knowledge acquisition in supervised learning networks," vol. 26, no. 6, nov 1996, pp. 801 –809.

[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, 2009.

[3] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, 2011.

[4] P. Chaovalit, "Clustering transient data streams by example and by variable," Ph.D. dissertation, Catonsville, MD, USA, 2009.

[5] P. P. Rodrigues, J. Gama, and J. P. Pedroso, "Hierarchical clustering of time-series data streams," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 5, pp. 615–627, 2008.

[6] Y. Chen, "Clustering parallel data streams," *Data Mining and Knowledge Discovery in Real Life Applications*, vol. I-Tech Education and Publishing, 2009.