ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

HACA: Hierarchical Aligned Cluster Analysis Using Temporal Boosting Clustering in Unsupervised Data

E.Santhiya[1], M.Praveena [2]

M.Phil Scholar, Dept. of Computer Science, Dr .SNS Rajalakshmi College of Arts & Science, Coimbatore, India[1].

Assistant Professor, Dept. of Computer Science, Dr .SNS Rajalakshmi College of Arts & Science, Coimbatore, India[2]

ABSTRACT: Temporal learning motion primitives as one of temporal boosting clustering, and derive an unsupervised hierarchical bottom-up framework called hierarchical aligned cluster analysis (HACA). HACA finds a partition of a given multidimensional time series into m disjoint segments such that each segment belongs to one of k clusters. Although HACA has shown promising results, there are a number of limitations. First, the computational complexity of ACA is $O(n^2 n_{max})$, which limits its applicability to long sequences. The proposed method of extending the HACA using sub sampling techniques to start the clustering using a decimated version of the time series and propagate it to another layer with higher temporal scales. This enhancement will improve the computational complexity in space and time.

KEYWORDS: Clustering; Aligned cluster analysis; Hierarchal cluster; Boosting

I. INTRODUCTION

Boosting algorithms [2] are a class of ensemble methods that have repeatedly proved highly effective in the context o f supervised classification. A somehow less explored scenario is the use of boosting techniques for unsupervised classification, namely clustering. Recently, there have been a few attempts to extend the boosting approach to the clustering domain. Some authors have proposed to combine different clustering algorithms in a boosting-like framework. For instance [3] introduced weighted re-sampling of data points according to how reliably they are classified. In [4], a general iterative clustering algorithm is presented that combines several algorithms by keeping track both of point weights and of a membership coefficient for each pair of a point and a model. Point weights are updated by a Bregman divergence optimisation procedure very similar to Adaboost [1].

Boosting concerns itself with the problem of combining several prediction rules with poor individual performance into a highly accurate predictor. This is commonly referred to as combining a set of “weak” learners into a “strong” classifier. Adaboost, introduced by Yoav Freund and Robert Schapire [1], differsfrom previous algorithms in that it does not require previous knowledge of the performance of the weak hypotheses, but it rather adapts accordingly. This adaptation is achieved by maintaining a distribution of weights over the elements of the training set. Adaboost is structured as an iterative algorithm, that works by maintaining a distribution of weights over the training examples. At each iteration a new weak learner (classifier) is trained, and the distribution of weights is updated according to the examples it misclassifies; the underlying idea is to increase the importance of the training examples that are “difficult” to classify. The weights also determine the contribution of the each weak learner to the final strong classifier.

The rest of this paper is organized as follows. In Section 2 review the existing related work. The proposed models and descriptions are described in Section 3. Finally conclude the paper in Section 4.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

II. RELATED WORK

In [5] authors discussed the classification involving imbalanced datasets has received considerable attention. Most classification algorithms tend to predict that most of the incoming data belongs to the majority class, resulting in the poor classification performance in minority class instances, which are usually of much more interest. In this paper we propose a clustering-based subset ensemble learning method for handling class imbalanced problem. In the proposed approach, first, new balanced training datasets are produced using clustering-based under-sampling, then, further classification of new training sets are performed by applying four algorithms: Decision Tree, Naïve Bayes, KNN and SVM, as the base algorithms in combined-bagging.In [6] authors discussed a proof which shows that weak learnability is equivalent to linear separability with $\ell 1$ margin. The equivalence is a direct consequence of von Neumann's minimax theorem. Nonetheless, we derive the equivalence directly using Fenchel duality. We then use our derivation to describe a family of relaxations to the weak-learnability assumption that readily translates to a family of relaxations of linear separability with margin. This alternative perspective sheds new light on known soft-margin boosting algorithms and also enables us to derive several new relaxations of the notion of linear separability.In [7] authors utilized a boosting refers to the general problem of producing a very accurate prediction rule by combining rough and moderately inaccurate rules-of-thumb. Boosting Methods combine many weak classifiers to produce a committee. It resembles Bagging and other committee based methods. Many weak classifiers are combined to produce a powerful committee. Sequentially apply weak classifiers to modified versions of data. Predictions of these classifiers are combined to produce a powerful classifier. In [8] authors he fact that K-means was proposed over 50 years ago and thousands of clustering algorithms have been published since then, K-means is still widely used. This speaks to the difficulty in designing a general purpose clustering algorithm and the ill-posed problem of clustering. They provided a brief overview of clustering, summarize well known clustering methods, discuss the major challenges and key issues in designing clustering algorithms, and point out some of the emerging and useful research directions, including semi-supervised clustering, ensemble clustering, simultaneous feature selection during data clustering, and large scale data clustering.

III. PROPOSED ALGORITHM

A. *TEMPORAL CLUSTERING*

Temporal Clustering of human motion into different activities is a crucial step for understanding and building computational models of human activity. This project explores robust methods to temporally segment, into coherent temporal patterns, streams of human behaviour coming from motion capture data of several subjects. Several issues contribute to the challenge of this task. These include large variability in the temporal scale and periodicity of human actions, inter-personal differences in motion patterns, and the exponential nature of all possible action combinations.

Segmenting human motion into distinct actions is a highly challenging problem. From the motion analysis perspective, segmentation is difficult due to large stylistic variations, temporal scaling, and changes in physical appearance, irregularity in the periodicity of human motions and the huge number of actions and their combinations. From a semantic viewpoint, segmentation is inherently elusive and difficult because in the vast majority of cases it is not clear when a set of poses describes an action. For instance, punching with the left hand and punching with right hand can be different actions, but it might be also regarded as punching or even more general as boxing.

An outcome of the clustering process is the temporal segmentation of long video sequences into event subsequence's. Once the frame features have been clustered into coherent shape/appearance clusters, the goal is to group them into a set of dynamic align motion (sets of consecutive clusters that occur more than p times, where p is a user specified criterion). The temporal representation of the video sequence, we are ready to find temporal patterns of different lengths. The algorithm starts selecting long patterns (usually $8 - 9$ consecutive clusters) as templates. Then, it computes normalized correlation of each of the templates with the sequence. All the instances that have normalized correlation of 1 (i.e. same pattern as the template) are removed from the sequence. If the data is too noisy, smaller thresholds than 1 can be imposed. After that, the algorithm selects smaller templates (typically one cluster less), searches again for all instances of normalized correlation 1, and proceeds this way until all the frames have been searched.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

B. *DYNAMIC TIME ALIGNMENT CLUSTER (DTAC)*

A temporal clustering algorithm needs to define a distance between segments of different length. Ideally, this distance should be invariant to the speed of the human action. This section reviews the DTAK that extends dynamic time warping (DTW) to satisfy the properties of a distance.

A frequent approach to aligning time series has been Dynamic Time Wrapping (DTW). A known drawback of using DTW as a distance is that it fails to satisfy the triangle inequality. To address this issue, the DTAC. Given two sequences X =[x_1, … x_{nx}] Є IR^{dxnx} and Y =[y_1, … y_{nx}] Є IR^{dxnx}. DTAC computes the similarity using dynamic programming. DTAC uses the cumulative kernel matrix UЄIR^{nx*ny}, computed in a recursive manner as,

$$\tau(\boldsymbol{X},\boldsymbol{Y}) = \frac{u_{nxny}}{n_x + n_y}, \qquad u_{ij} = max\begin{cases} u_{i-1,j} + k_{ij} \\ u_{i-1,j-1} + 2k_{ij} \\ u_{i,j-1} + k_{ij} \end{cases} \qquad eqn.(1)$$

A more revealing mathematical expression to understand DTAK can be obtained using matrix notation.

C. *HIERARCHICAL ALIGNED CLUSTER ANALYSIS (ACA)*

The hierarchical ACA (HACA), a hierarchical extension of ACA. HACA reduces the computational complexity of ACA and provides a hierarchical decomposition at different temporal scales. ACA's computational cost is linear in the length constraint n_{max} and quadratic in the length of the sequence n. Unlike ACA, HACA starts the search with small temporal scales and propagates the result to larger temporal scales. The computational complexity is O($n^2 n_{max}$) and HACA is more efficient because it starts using smaller temporal scales (i.e., n_{max}). For instance, if the length constraint for ACA is n_{max}, the equal setting for a two-level HACA involves two pairs, n_{max}(1) and n_{max}(2), where n_{max}(1) and n_{max}(2) denote the constraints used in the first and second level, respectively. In the following, we show how to compute HACA using the same algorithm as ACA, but replacing the kernel DTAK with the generalized DTAK (GDTAK). The decision tree algorithms it has been proposed to use dynamic sub-sampling at each node in order to determine the optimal test. A Dynamic Programming Piecewise Linear Approximation model is used to automatically extract in an optimal way key-postures distributed along the motion data. This non uniform sub-sampling can be exploited for motion compression, segmentation, or re-synthesis. The adaptive luma sub sampling based on a combination of three perceptually significant image characteristics (texture, edges, and brightness) to complement uniform chroma sub sampling.

D. *ENERGY FUNCTION FOR ACA*

There are two major challenges in framing temporal segmentation as a clustering problem: (1) modelling the temporal variability of human actions, and (2) defining a robust metric between temporal actions. To address these problems, ACA extends previous work on kernel k-means by minimizing:

$$J_{ACA}(G,s) = \sum_{c=1}^{k}\sum_{i=1}^{m} g_{ci}\,dist_c\left(X_{[si,si+1]}\right) \qquad eqn.(2)$$

It is worth pointing out the differences between ACA, and kernel k-means (1) ACA clusters variable features, that is, each segment Y_i might have a different number of frames, whereas standard kernel k-means has fixed number of features (rows). (2) The kernel used in ACA, dist c(Y_i), uses DTW that is robust to noise and invariant to the speed of the action.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

IV. CONCLUSION AND FUTURE WORK

The HACA is an extension of kernel k-means for temporal clustering. ACA combines standard vector-space approaches for clustering with Dynamic Time Alignment Clustering (DTAC) and Dynamic Time Wrapping (DTW). The main contributions of this paper are: (1) formulation of temporal clustering with ACA, (2) temporal reduction and initialization strategies for ACA, and (3) efficient computation of using hierarchical techniques to start the clustering using a decimated version of the time series and propagate it to another layer with higher temporal scales. Although HACA has shown promising preliminary results, there is still the need for algorithms to automatically select the optimal number of actions and avoid local minima in the optimization. This enhancement will improve the computational complexity in space and time.In the future, it is likely that wavelet processing will be common in video processing systems, and it will be straightforward to utilize this range of various actions compression.

REFERENCES

1. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application toboosting. Journal of Computer and System Science 55(1) (1997)
2. Freund, Y., Schapire, R.: A short introduction to boosting. Journal of Japanese society for Artificial Intelligence Science 14(5) (1999)
3. Frossyniotis, D., Likas, A., Stafylopatis, A.: A clustering method based on boosting. Pattern RecognitionLetters 25, 641–654 (2004)Nock, R., Nielsen, F.: On weighting clustering. IEEE Transactions on PAMI 28(8), 1223–1235 (2006)
4. Topchy, A., Minaei-Bidgoli, B., Jain, A.K., Punch, W.F.: Adaptive clustering ensembles. In: Proc. 17th Conf. Pattern Recognition, pp. 272–275 (2004)
5. H. Xiong, Y. Yang, and S. Zhoa, "Local clustering ensemble learning method based on improved adaboost for rare class analysis," J. Comput. Inform. Syst., vol. 8, pp. 1783–1790, 2012.
6. S. Shalev-Shwartz and Y. Singer, "On the equivalence of weak learnability and linear separability: New relaxations and efficient boosting algorithms," Mach. Learn., vol. 80, pp. 141–163, 2010.
7. A. Ganatra and Y. Kosta, "Comprehensive evolution and evaluation of boosting," Int. J. Comput. TheoryEng., vol. 2, pp. 931–936, 2010.
8. A. Jain, "Data clustering: 50 years beyond K-means," Pattern Recog. Lett., vol. 31, pp. 651–666, 2010.