ONLINE LEARNING FOR IMAGE PROCESSING IN NETWORKED SETTING

ANKUSH RAI*, JAGADEESH KANNAN R

School of Computing Science & Engineering, VIT University, Chennai, Tamil Nadu, India. Email: ankushressci@gmail.com

ABSTRACT

Objective: In the past decade, development of machine learning algorithm for network settings has witnessed little advancements due to the slow development of technologies for improving bandwidth and latency.

Methods: In this study, we present a novel online learning algorithm for network-based computational operations in image processing setting. We take a different route and focused on creating an unsteady checkpoint based system which reacts to the sequences in a constructive and destructive manner. While the past studies are focused on discarding instances to keep the bounded state of the support set

Results and Discussion: The main advantage posed by the above algorithm is the setup it put forth for scheduling training and processing based on memory bound to the support set on timely basis and thus forming a decomposable or expanding sequence when the other instance pairs are added in relational to the previous trained hypothesis, such that the trained hypothesis is always bounded and deducible from the other previous pairs of instances. The training is achievable in small number roof instances with high accuracy.

Keywords: On-line learning, Image processing over network, Hypothesis deduction.

INTRODUCTION

The learning algorithms based on kernels such as support vector machines, Gaussian process, regularization networks are proven its usefulness in many batch settings processing environment [1]. However, for developing an online learning algorithm one can implement it by using the sliding buffer in combination with batch algorithms [2]; but the extension of kernel learning methods to online learning method has its own unsolved challenges which inhibit its real-time applicability in the scenario of data services. The challenges for online learning algorithms are as follows:

1. When applied to Hilbert space method due to the high dimensionality of the weight vectors the standardized online settings for those corresponding to linear methods are often prone to overfitting. This can manage by the effective utilization of prior probabilities in the function space by regularizing it.

2. Second, as the number of observations increases it becomes more complex to functionally represent the classical kernel estimators; according to the represented theorem, there will be linearly increase in a number of kernel functions with the total number of observations [3]. As it is prone to happen more often while depending on loss function used [4]. Therefore, there is an increase in the complexity of the overtime with higher storage requirements [5,6].

3. Third, the training time required for a batch or for an incremental update algorithm will always increases super-linearly with observations. Although incremental update algorithms tried to overcome the problem, still cannot prove its ground over bounds of operations required per iterations [7]. Whereas, projection algorithms tries to limit the number of updates per iteration and thereby ensuring the complexity to remain constant. Consequently, they are computationally expensive and required multiplication operations at each of its steps [8-10].

In the following sections, we present the new approach to tackle such problem and have implemented the system over an online image restoration process as a testing scenario to gauge the affectivity of the algorithm. Section 2 includes the model and the algorithm, whereas the section 3 includes the results and conclusions of the study.

METHODOLOGY: SEQUENTIAL INCREMENTAL ONLINE LEARNING ALGORITHM (SIOLA) WITH RISK ESTIMATOR

The experimental setup for the application and testing of the proposed algorithm is shown in Fig. 1. After that, the algorithm is executed to run over the network to learn the data slip and latter recover the missed data through the proposed SIOLA online learning algorithm. Unlike other algorithm, we take a different route and focused on creating an unsteady checkpoint based system which reacts to the sequences in a constructive and destructive manner. While the past studies are focused on discarding instances to keep the bounded state of the support set, instead we employ a sequence based fusion techniques that reacts to the pattern in a way which can let it build itself the past instances by keeping the biasing and computing through the sequences to dynamically create a support set and thereafter adjoining the essential part of it which latter shall help develop the same checkpoints by the application of the similar sequence [11-13]. The variation in the hypothesis is encoded and decode through a sequence which is based on the trading equilibrium of sparseness. The algorithm described below as follows:

Algorithm: SIOLA

Input: s_i, s_j are the time instances of the support pairs, S_t is the support set, H_t is the online hypothesis at time t and p_t is the pair equilibrium sequence.

Output: H_o is the online hypothesis and updated p'_t.

Step 1: For t = 1, 2, 3 ….

Step 2: Receive new instance S_t

Step 3: Attach time instances of the pairs.

$$p_t = \frac{[s_i * s_j * q]}{[s_i * s_j]} \quad \text{//paired equilibrium sequence.}$$

$$S_t = \sum_i p_t(t - t_i) \text{ //Binding process.}$$

Step 4: Evaluate the risk of pairing and remaining time instance of memory bound.

While $k_i \neq k_n$

{

$$R[k_i] = \frac{1}{p_t} \sum_{l=1}^{t} (S_t - S_{t-1}) \int_t k_i dt$$

$$M_t = \sum_k k(t - t_i) \text{ Where,}$$

$$M_t = \Sigma_k k(t - t_i)$$

$$k_i = \frac{(p_t \, \Delta t)^k}{k!} \exp(-p_t \cdot \Delta t)$$

Step 5: Update the hypothesis with chain sequence by checking for risks involved.

if $R[k_i] \leq R[S_t]$

{

$$\sum_{k=l}^{\infty} \sum_{k_i}^{m} \frac{(p_t)^{k-k_i}}{(M_t - t_i)!} \cdot \exp(-p_t \cdot \Delta t) \cdot \frac{(p_t)^{k_i}}{(k_i)!}$$

}

else

{

Print "failed to update."

k_i = Null//Assign

$k_i = k_{i+1}$

}

Step 6: Add instances to the support set.

if $p_t t_i \leq k_i$

{

$$\sum_{k=l}^{\infty} \frac{(p_t t_i + p_{t+1} t_{i+1})^k}{k!} \exp(-p_t \cdot \Delta t)$$
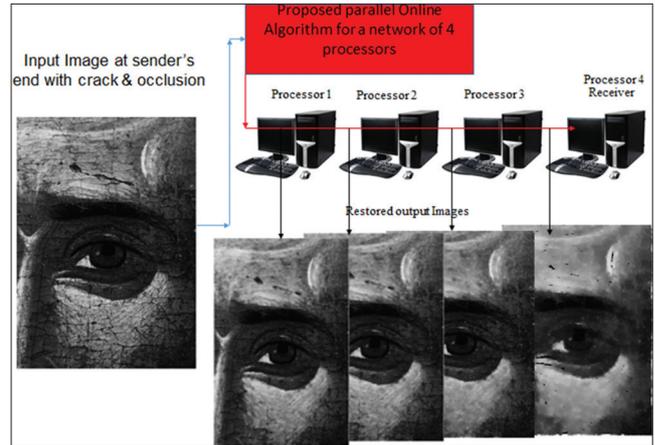
}

else

{

$S_t = S_{t-1}$

}

Fig. 1: Experimental setup of the image transmitted through four sequentially arranged data processors using anisotropic image transformation algorithm [9]. The respective restored output images during each phases of the online hypothesis so generated are shown above
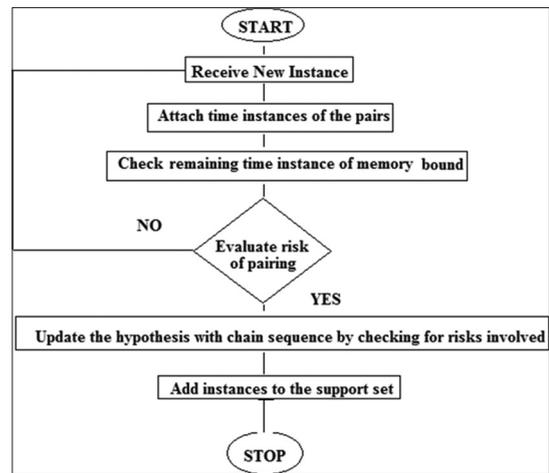

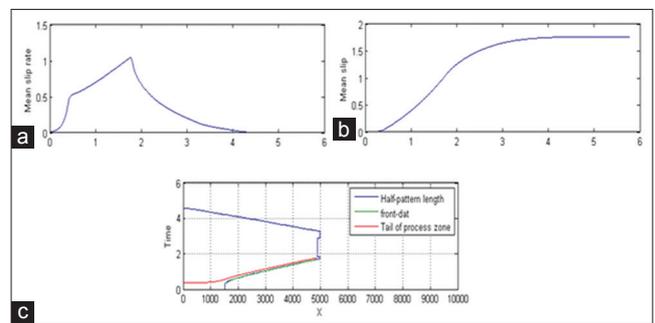**Fig. 2: Flow chart of the proposed sequential incremental online learning algorithm algorithm**

Fig. 3: (a) The mean slip rate v/s time in seconds for the first phase in flow of data transmission, (b) Mean slip rate v/s time in seconds for the end phase in flow data transmission achieving saturation at converge level, (c) plot of the time v/s flow and the transmitted data (x); giving rise to curvelet transformation for the half pattern length of sequences used in the proposed algorithm

}//end while.

Step 7: End process.
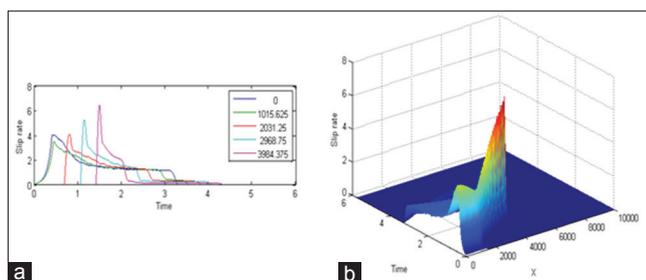
Fig. 4: (a) Data slip rate for the network flow during the data transmission, (b) 3D plot of the data slip rate in combination with the time flow and the transmitted data (x)

The main advantage posed by the above algorithm is the setup it put forth for scheduling training and processing based on memory bound to the support set on timely basis and thus forming a decomposable or expanding sequence when the other instance pairs are added in relational to the previous trained hypothesis, such that the trained hypothesis is always bounded and deducible from the other previous pairs of instances. The training is achievable in small number roof instances with high accuracy. The workflow chart of the proposed algorithm is given in Fig. 2. Here, the Fig. 3a represents the onset of the data slip rate due to the bounded memory units to update the online hypothesis. However, the by the end of the recovered image the data slip rate haven't shown any fluctuation in loss of data, thereby converging the results at the saturation level with respect to the time (Fig. 3b). The advantage that our online learning algorithm put forth is its adaptability in re-organizing the online hypothesis so generated during the process run. This lead to a curvelet transformation for the process to end at the point where the front data and the half pattern length saturates (Fig. 3c). The process is cyclic in nature and does not require unnecessary updation in due process.

CONCLUSIONS

The average online error of the proposed SIOLA algorithm is a function directly dependent on the size of support sets for different data sets. This algorithm solves the challenges with online learning by computationally proving convergence rates and error bounds in the size of the support sets with descent in the number of the mistakes made which gradually saturates at nominal level (Fig. 4). We summarize the contributions of this study as a new algorithm for online learning algorithm with a dynamically bounded support set and an extension of the exemplified previous online learning technique (Fig. 4). Here, the size of the solution is guaranteed to be bounded by the SIOLA to solve the memory explosion problem. In addition, the size of the support set

is adaptable with the memory bound units thus the size of the support sets is comparatively smaller than the budgets based learning methods. This allows the applications of to rely over online based processing methods such as image processing, novelty detection for online and time-varying problems. Our works solve the online to batch conversion to a bounded batch solution, which is scalable in the quadratic size of the support sets. The result put forth in the study provides the essential theoretical insights in the scenario of drifting targets and dynamic hypothesis.

REFERENCES

1. Bennett KP, Mangasarian OL. Robust linear programming discrimination of two linearly inseparable sets. Optim Methods Softw 1992;1(1):23-34.
2. Cauwenberghs G, Poggio T. Incremental and decremental support vector machine learning. In: Leen TK, Dietterich TG, Tresp V, editors. Advances in Neural Information Processing Systems. Vol. 13. Cambridge, MA: MIT Press; 2001. p. 409-15.
3. Csat´o L, Opper M. Sparse representation for Gaussian process models. In: Leen TK, Dietterich TG, Tresp V, editors. Advances in Neural Information Processing Systems. Vol. 13. Cambridge, MA: MIT Press; 2001. p. 444-50.
4. Friedman J, Hastie T, Tibshirani R. Additive Logistic Regression: A Statistical View of Boosting. Technical Report, Stanford University, Department of Statistics; 1998.
5. Gentile C. A new approximate maximal margin classification algorithm. In: Leen TK, Dietterich TG, Tresp V, editors. Advances in Neural Information Processing Systems. Vol. 13. Cambridge, MA: MIT Press; 2001. p. 500-6.
6. Graepel T, Herbrich R, Williamson RC. From margin to sparsity. In: Leen TK, Dietterich TG, Tresp V, editors. Advances in Neural Information Processing Systems. Vol. 13. Cambridge, MA: MIT Press; 2001. p. 210-6.
7. Guo Y, Bartlett P, Smola AJ, Williamson RC. Norm-based regularization of boosting. Submitted J Mach Learn Res 2001:1001-22.
8. Herbster M. Learning Additive Models Online with Fast Evaluating Kernels. In: Proceedings. 14th Annual Conference on Computational Learning Theory (COLT). Springer; 2001. p. 444-60.
9. Perona P, Malik J. Scale-space and edge detection using anisotropic diffusion. IEEE Trans Pattern Anal Mach Intell 1990;12(7):629-39.
10. Rai A, Ramanathan S. Distributed learning in networked controlled cyber physical system. Int J Pharm Technol 2016;8(3):18537-46.
11. Ankush R. Application of Artificial Intelligence for Virtually Assisted Prognosis of Diabetes: A Nodds Project. IJCA Proceedings on National Seminar on Application of Artificial Intelligence in Life Sciences. NSAAILS(1):1-5, February; 2013.
12. Rai A, Ramanathan S, Kannan RJ. Quasi Opportunistic Supercomputing for Geospatial Socially Networked Mobile Devices. Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2016 IEEE 25th International Conference on IEEE; 2016.
13. Rai A. Unsupervised Probabilistic Debugging. Recent Trends Program Lang 2015;1(2, 3):14-6.