© 2015 IJSRSET | Volume 1 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Parallel Multitasking In Real Time Applications

Ramya A, Ramya B, Suryaa K
Dhanalakshmi College of Engineering, Chennai, Tamilnadu, India

ABSTRACT

Parallel multitasking is mainly for solving multiple related classifications of tasks parallel. It classifies every sequence of data received by each task accurately and efficiently. Parallel multitasking also enables parallel micro-blogging on a group of users, which allows the user to post views as well as see the views posted by others. It maintains a global model over the entire data of all tasks. The individual models for multiple related tasks are jointly inferred by leveraging the global model through a parallel multitasking approach. The user can work on multiple applications at the same time. Experimental results show that the method is effective and scalable at the online classification of multiple tasks.
Keywords: Artificial intelligence, learning systems, online learning, multitask learning, classification

I. INTRODUCTION

Classical methods for learning are often as single task learning. The standard methodology in machine learning is to learn one task at a time. Large problems are divided into small independent sub problems that are learned separately and then they are recombined. This methodology is counter-productive because it ignores potentially rich source of information available in many real-time problems. The information contained in the training signals of other tasks drawn from the same domain.

Many real-world problems are essentially multitask learning, although they are often broken into smaller single learning tasks, which are then solved individually by classical learning methods. The classical multitask learning methodology often makes two assumptions. First, it assumes there is one primary task and other related tasks are simply secondary ones whose training data are exploited by multitask learning to improve the primary task.

Thus, the classical multitask learning approach focuses on learning the primary task without caring how the other tasks are learned. Second, the classical multitask learning problem is often studied in a batch learning setting or in offline setting, which assumes that the training data of all tasks are available. It is not suitable for many real-time problems where data arrives sequentially. On the other hand, the batch learning algorithms usually have intensive training cost and poor performance.

In this paper, we investigate the problem of parallel multitasking. Our goal is to improve the learning performance of all tasks instead of focusing on a single primary task. Unlike batch learning methods, online learning methods learn over data by processing each sample accurately upon arrival.

II. METHODS AND MATERIAL

Related Work

Our work is closely related to two groups of research in machine learning and data mining, i.e., (1) online learning, and (2) multitask learning. We briefly survey the representative work in each area .our technique jointly learns a generic global model shared by many parallel learners and individual collaborative models by reinforcing each learning task through a collaborative learning process. It improves the classification performance, but also retains the hallmark low computational cost of online learning algorithms.

Formulation and Algorithms

We now formulate the problem in a binary classification setting. The proposed algorithm can be easily extended to address the multiclass problems. Parallel multitask classification proceeds in rounds by observing a sequence of samples, each from some user/task from a set of K users/tasks.

On each round, there are K separate online binary classification problems being solved jointly. We assume that data from all users/tasks can be represented in the same global feature space, so that we adopt the online passive aggressive (PA) framework to build a global model using data collected from all users at round t, that is

$$f_t(\mathbf{x}) = \mathrm{sign}(\mathbf{u}_t \cdot \mathbf{x})$$

Where $\mathbf{u}_t \in R^d$ is the weight vector of the global model learned at round *t*. specifically, at round *t*, the algorithm uses the latest training instance _$\mathbf{x}_t$, y_t_ to update the classification model

III. RESULTS AND DISCUSSION

Implementation

We implement this project with varied real time applications. Here we consider three real life applications of spam mail filtering, multitasking real time applications and parallel micro blogging.

Spam Email Filtering

We apply parallel multitasking to construct effective personalized spam email filters. The task is to classify each new incoming email message into two categories: legitimate or spam.

The set of all emails received by a user is not generated by that specific user. However, the characteristic of each user's email can be said to match his or her interest.

Each email entry will be converted to a word document vector using the TF-IDF (term frequency-inverse document frequency) representation. Since the email dataset has no time stamp, each user's email was shuffled into a random sequence.

It should be noted that compared to online learners who update models based only on the current sample, batch learning methods have the advantage of keeping a substantial amount of recent training samples, at the cost of storage space and higher complexity. In fact, the proposed COML algorithm is more efficient than batch incremental COML does not store recent training samples. It only uses the current training sample and a simple rule to update the model. Batch learning algorithms need to keep a certain number of recent training samples in memory, leading to extra burden on storage and complexity.

Parallel Micro blog

Micro blogging is a broadcast medium that exists in the form of blogging. A micro blog differs from a traditional blog in that its content is typically smaller in both actual and aggregated file size. Each and every blogs posted by user will be stored in database.

Micro blogging is all about posting blogs on the web where User able to post blogs and User can able to view the blogs. Here parallel micro blogging is done where both the things will run parallel. All the blogs posted by user will be monitored by admin.

Here admin is present, who co-ordinates and manages the parallel microblogging system. The admin equally has the facility to post blogs and can able view all the posted blogs. The admin also analyse the blogs in terms of name, type and the blog posted.

Multitasking Real Time Applications

The user can able to view and perform multiple applications on a single page. Based on the user input applications will be displayed .Each and every application will run parallel. Easy for users to learn in a multiple tasks. One application will not disturb another application. At the same time displaying multi applications on a single page will not affect the performance as well. Hence the user can work on one application and can simultaneously view the updates from other applications. For example. a user is shopping goods online can see the relied offers to the purchase in another application. This method makes learning faster and more accurate.

IV. CONCLUSION

In this paper, we proposed parallel multitasking method that is used to gain individual and global models to achieve improvement in performance of combined learning of various related tasks. The proposed system is able to perform by integrating them in a unified parallel learning framework. The experimental results demonstrate that our algorithms are both effective and efficient for three real-time applications, including online spam email filtering, multitasking real-time applications and parallel micro-blog Although the parallel multitasking was firstly designed to solve the UGC defining problem, it has potential applications outside of the domains studied here. We hope to be able to extend. Our experiments to a more substantial size dataset and also to more applications. Our methods assume uniform relations across tasks. However, it is more reasonable to take into account the degree of relatedness among tasks. How to incorporate hierarchies and clusters of tasks is also worthy of further study.

V. REFERENCES

[1] R. Caruana, “Multitask learning,” Mach. Learn., vol. 28, no. 1, pp. 41–75, Jul. 1997.

[2] T. Evgeniou, C. A. Micchelli, and M. Pontil, “Learning multiple tasks with kernel methods,” J. Mach. Learn. Res., vol. 6, pp. 615–637, Apr. 2005.

[3] G. Li, S. C. H. Hoi, K. Chang, and R. Jain, “Micro-blogging sentiment detection by collaborative online learning,” in IEEE 10th ICDM, Sydney, NSW, Australia, 2010, pp. 893–898.

[4] G. Li, K. Chang, S. C. H. Hoi, W. Liu, and R. Jain, “Collaborative online learning of user generated content,” in Proc. 20th ACM Int.CIKM, 2011, pp. 285–290.

[5] A. Saha, P. Rai, H. Daumé, III, and S. Venkatasubramanian, “Online learning of multiple tasks and their relationships,” in Proc. 14th Int. Conf. AISTATS, Ft. Lauderdale,FL, USA,2011,pp. 643–651.

[6] G. Cavallanti, N. Cesa-Bianchi, and C. Gentile, “Linear algorithms for online multitask classification,” J. Mach. Learn. Res., vol. 11, pp. 2901–2934, Oct. 2010.

[7] J. Zhou, J. Chen, and J. Ye, (2011). MALSAR: Multi-task Learning via StructurAl Regularization, Arizona State University, [Online]. Available: http://www.public.asu.edu/~jye02/Software/MALSA

[8] S. Ben-David and R. Schuller, “Exploiting task relatedness for mulitple task learning,” in 16th Annu. COLT, Washington, DC, USA, 2003, pp. 567–580.

[9] E. V. Bonilla, K. M. Chai, and C. K. I. Williams, “Multi-task Gaussian process prediction,” in NIPS, Dec. 2007.

[10] D. Sheldon, “Graphical multi-task learning,” in NIPS’08 Workshop Structured Input Structured Output, 2008.

[11] R. K. Ando and T. Zhang, “A framework for learning predictive structures from multiple tasks and unlabeled data,” J. Mach. Learn. Res., vol. 6, pp. 1817–1853, Nov. 2005.

[12] O. Chapelle, P. K. Shivaswamy, S. Vadrevu, K. Q. Weinberger, Y. Zhang, and B. L. Tseng, “Multi-task learning for boosting with application to web search ranking,” in Proc. 16th ACM SIGKDD Int. Conf. KDD, Washington, DC, USA, 2010, pp. 1189–1198.

[13] Z. Kang, K. Grauman, and F. Sha, “Learning with whom to share in multi-task feature learning,” in Proc. 28th ICML, Bellevue, WA, USA, 2011.

[14] A. Argyriou, C. A. Micchelli, M. Pontil, and Y. Ying, “A spectral regularization framework for multitask structure learning,” in NIPS, 2008, pp. 25–32.

[15] L. Jacob, F. Bach, and J. Vert, “Clustered multi-task learning,” in NIPS, 2009, pp. 745–752.

[16] K. Q. Weinberger, A. Dasgupta, J. Langford, A. J. Smola, and J. Attenberg, “Feature hashing for large scale multitask learning,” in Proc. 26th Annu. ICML, 2009, p. 140.

[17] O. Dekel, P. M. Long, and Y. Singer, “Online multitask learning,”in 19th Annu. COLT, Ft. Lauderdale, FL, USA, 2006, pp. 453–467.