Continuous Clustering in Big Data Learning Analytics

Kannan Govindarajan[1] , Thamarai Selvi Somasundaram[1], Vivekanandan S Kumar[2] , Kinshuk[2]

[1]Madras Institute of Technology,
Anna University, Chennai, India
Email: kannan.gridlab@gmail.com
stselvi@annauniv.edu,

[2]Athabasca University,
Edmonton, Canada
Email: vive@athabascau.ca,
kinshuk@athabascau.ca

Abstract— **Learners' attainment of academic knowledge in post-secondary institutions is predominantly expressed by summative or formative assessment approaches. Recent advances in educational technology has hinted at a means to measure learning efficiency, in terms of personalization of learner competency and capacity in terms of adaptability of observed practices, using raw data observed from study experiences of learners as individuals and as contributors in social networks. While accurate computational models that embody learning efficiency remain a distant and elusive goal, big data learning analytics approaches this goal by recognizing competency growth of learners, at various levels of granularity, using a combination of continuous, formative and summative assessments. This study discusses a method to continuously capture data from students' learning interactions. Then, it analyzes and clusters the data based on their individual performances in terms of accuracy, efficiency and quality by employing Particle Swarm Optimization (PSO) algorithm.**

Keywords: Big Data, Learning Analytics, Particle Swarm Optimization (PSO)-based Clustering, Hadoop, K-Means Clustering.

I. INTRODUCTION

The analysis and discovery of relations between human learning and contextual factors that influence these relations have been one of the contemporary and critical global challenges facing researchers in a number of areas, particularly in Education, Psychology, Sociology, Information Systems, and Computing. Traditionally, these relations concern learner performance and the effectiveness of the learning context from summative and formative points of view. Be it the assessment marks distribution in a classroom context or the mined pattern of best practices in an apprenticeship context, analysis and discovery have always addressed the elusive causal question about the need to best serve learners' learning efficiency. Learning efficiency encompasses any and all aspects that concern "learning" of individual learners or groups of learners. Examples of learning efficiency aspects include learning style, metacognitive scaffolds, peer interactions, self-regulation, co-regulation, social networking, and other learning-oriented activities and characteristics associated with learners. With the advent of new technologies such as eye-tracking, activities monitoring, video analysis, content analysis, sentiment analysis and interaction analysis, one could potentially collect "continuous data", in addition to formative and summative data. Continuous data is different from the other two in terms of its incessant arrival from direct observations of a learning activity and other activities related to that learning activity. For example, assessment of a submitted essay by a student offers summative data. Observing the development of the essay that assists the learner (or the teacher) in making targeted decisions about the quality of the essay being written offers formative data. This data can be obtained at real time and can be used to classify each student's progress in a learning task and to develop a model of growth of competency, say in writing essays. The volume and arrival rate of continuous data leads to big data learning analytics.

Big data analytics, as opposed to smaller data analytics that can be equated to approaches that use data mining or simpler artificial intelligence in education techniques, targets large volumes of data (beyond terabytes) as well as large numbers of voluminous computational models (models using a significant number of variables) concerning learning efficiency. Big data is characterized using the following five factors – volume, speed, variety, veracity, and value. Learning Analytics is different from Artificial Intelligence in Education in terms of the focus on learning evolution. Learning Analytics is different from Educational Data Mining in that it does not expect well-defined data to be available in a repository. Learning Analytics is different from User Modelling on the use of big data as the basis. It can be readily applied in the domains of reading, writing, free-hand writing, coding, mathematical problem solving, understanding learning styles, gaming, chats (e.g., video, audio, text), in-class performances, metacognitive activities (e.g., self-regulation, co-regulation), social network contributions (e.g., social network analysis), and usage of learning resources/tools such as Matlab, SPSS, Eclipse, Moodle and Cognitive Authoring Tools (CTAT).

There are many techniques available to analyze student competencies based on big data on learning performances. One such technique concerns clustering of students based on their performances to date on a learning activity. Clustering is the process for grouping of similar data objects, for instance, students with similar misconceptions could be clustered. Clustering algorithms are classified into two types such as supervised and unsupervised learning. The supervised learning algorithm has an external director for making decision to form the clusters. The unsupervised learning algorithm works on the principle of finding similarity or distance between the objects and forming of clusters based on the similarity measures. There are various clustering algorithms such as K-Means, Centroid Clustering, Fuzzy-means and so on. The main drawback of the existing clustering algorithms is the random selection of initial centroids and their ability to deal with continuous arrival of data. This paper presents a novel Particle Swarm Optimization (PSO) based unsupervised clustering algorithm for clustering of students for continuously arriving data. It is a population-based stochastic optimization technique, which has been modeled based on the swarm of particles. The particles in each

978-0-7695-5141-8/13 $31.00 © 2013 IEEE
DOI 10.1109/T4E.2013.23

swarm represent the potential solution. Each particle in the swarm tries to find the optimal solution by considering task, emotive, social and cognitive factors. The algorithm utilizes a cloud based storage resources for storing big data. The paper will describe the PSO algorithm, use the algorithm to cluster students based on 3 sample factors – efficiency, accuracy and error count, and use simulated data to assess the performance of the algorithm. Section II describes the background and related work. Section III describes the system architecture. Section IV discusses the PSO algorithm. Section V discusses the study and Section VI concludes the paper.

II. RELATED WORKS

There has been a lot of research effort carried out in monitoring and discovery and we discuss some of the works that has been closely related to our work A surge of interest in educational data mining has been observed in recent years [1-5]. Kadir Geyik [6] proposed the concept of clustering the learners based on the attributes such as recently accessed materials, frequently accessed materials and monetary factors, where a hierarchical clustering algorithm clustered learners based on self-organizing map, followed by a non-hierarchical clustering approach called fuzzy clustering. Stavros Valsamidis [7] used the Markov clustering algorithm to cluster learners based on their activities with respect to enrichment, interest and disappointment. Xiaohui Cui et al. [8] proposed Particle swarm optimization based clustering to cluster documents using cosine similarity as a measure to cluster similar documents. Dheeban S.G. et al. [9] proposed a personalized learning approach using Modified Particle Swarm Optimization, where learners are grouped based on their ability level and difficulty of the course, thus allowing an estimation of a course's suitability for a learner. Sridevi et al. [10] proposed cosine semantic similarity measure to cluster annotated documents using PSO based clustering. In all these approaches, initial solutions have been randomly chosen and hence offer less-than-optimal convergence. Further, these approaches do not offer optimal clustering solutions when the data tends to arrive continuously. To address these two issues, one would require an algorithm that approaches convergence much faster than contemporary approaches. Further, such an algorithm would have to inform intermediate solutions about newly arrived data sets in order to revise these intermediate solutions on a continuous basis. The novel PSO algorithm proposed in this paper addresses these two key concerns.

III PROPOSED SYSTEM ARCHITECTURE

The system-level architecture for implementing our proposed work is shown in Figure 1. The system fetches three types of data from students' study activities in the domain of Java programming - exercises, assignments and assessments. These three data sets are preprocessed before being fed into the PSO-clustering manager. We developed a Hackystat based continuous data trace mechanism to collect coding related data. This mechanism continuously collected data from Eclipse as and when students engaged in UML-based design of Java programs, writing of programs, debugging of programs, documenting programs, and testing programs. The Coordinator is the entry point for the proposed system that acts as the mediator for coordinating various activities such as (a) Invoke the Data Collector to collect or fetch the continuous data (b) Invoke the Data Preprocessor to process the collected data (c) Invoke the PSO-based Clustering Manager to cluster the students based on the similarity value (d) Invoke the Resource Allocator to select the cloud resources for storage activity c) Invoke the Cloud Resource Information Aggregator in a periodic interval to dynamically aggregate the physical resource information about Cloud resources (e) Invoke the Data Storage Manager for efficiently storing the structured data in Hadoop Distributed File System (HDFS). The Data Preprocessor processes the raw data; the processed structured data is fed into PSO-based clustering manager for further processing. It includes an analysis of students' discussion forum contributions as well as sentiment analysis input obtained from students. Data Collector retrieves the student's data from the tools such as Moodle Learning Management Systems (LMS), Virtual Programming Lab (VPL), Intelligent Tutors, and Eclipse IDE extensions. The Cloud Resource Information Aggregator makes use of Zookeeper to monitor and manage the Hadoop clusters. The Resource allocator is the component that helps to locate suitable resources for data storage as well as data retrieval. The Data Storage Manager is responsible for interacting with Hadoop Distributed File System (HDFS) to efficiently store and retrieve the unprocessed data as well as processed data.

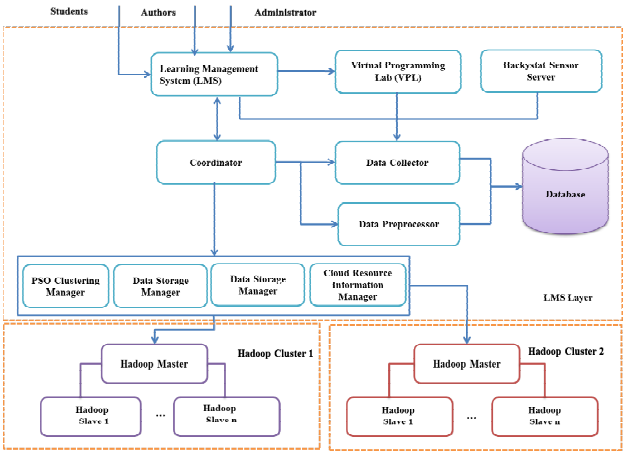

FIGURE 1: PROPOSED SYSTEM ARCHITECTURE

IV PARTICLE SWARM OPTIMIZATION (PSO) CLUSTERING

Particle Swarm Optimization (PSO) is a population-based artificial intelligence mechanism that is inspired by social behavior of swarm of birds. PSO algorithm is employed with the swarm of particles, and every particle is responsible for tracking the fitness of each particle. Every particle is associated with corresponding velocity that helps the particle to move onto best position and every particle in the swarm search the better solution. The convergence of PSO depends on the particle's personal position and global best position of swarm. The number of students to be clustered is M. In the PSO clustering algorithm, a swarm of particles P, each driven with a velocity V, aims to cluster the learners according to

their observed behaviors. The particle 'P' is represented as, $P_i \in P$ have 'D' dimensions. In PSO clustering, each dimension represents a cluster centroid. In other words, the number of clusters to be formed is 'D'. The characteristic of a student $S_j \in S$ is given as

$$S_j \leftarrow \left\{ SID_{S_j}, Accuracy_{S_j}, Efficiency_{S_j}, Quality_{S_j} \right\}.$$

The pseudo code for PSO-based clustering algorithm is illustrated under Algorithm 1 shown below:

Algorithm 1: Particle Swarm Optimization (PSO)

Input: Number of particles $P \in \{P_1, P_2, ,, ,..., P_P\}$

A particle $P_i \in P$ is represented as $(ce_1, ce_2, ..., ce_D)$ where ce_i represents centroid of the cluster 'i'. Cluster centroid ce_i is represented as $ce_i = (x_1, x_2, ...x_N)$ where N denotes the number of attributes or features.

Output: An optimal particle P_i that represents cluster centroid

For each particle $P_i \in P$

Initialize $Pbest_{P_i \in P}$ as ∞

Initialize $Gbest_{Clus}$ as ∞

Initialize $PbestPos_{P_i \in P}$ as $(0_1, ...0_k)$

Initialize $GbestPos_{Clus}$ as $(0_1, ...0_k)$

Initialize velocity of $V_{P_i \in P}$ as $rand \in_U \{0,1\}$

Initialize particle $P_i \in P$ as $(ce_1, ce_2, ..., ce_D)$

End

Repeat until max generation

/* calculating fitness function */

For each particle $P_i \in P$

For each student $S_j \in S$ where 1 < j < M

For each cluster centroid $ce_t \in P_i$

Where $1 \le t \le D$

$$f(P_i) = \sum_{j=1}^{M} \sum_{t=1}^{n} d(s_j, ce_t)$$

End

End

End

/* choosing Pbest and Pbest position */

For each particle $P_i \in P$

If $Pbest_{P_i \in P} > f(P_i)$

$Pbest_{P_i \in P} \leftarrow f(P_i)$

$PbestPos_{P_i \in P} \leftarrow (ce_1, ce_2, ..., ce_K)_{P_i}$

End

End

/* choosing Gbest and Global best Position */

$$Gbest_{Clus} \leftarrow \min\left\{ Pbest_{P_i \in P} \right\}$$

$$GbestPos_{Clus} \leftarrow (ce_1, ce_2, ..., ce_K)_{P_i}$$

/* updating the velocity and position * /

For each particle $P_i \in P$

Update the velocity and position for the next iteration

End

End (of max generations).

V EXPERIMENTAL SETUP AND RESULTS

A. Experimental Setup

To evaluate the suitability of the PSO algorithm for big data learning analytics, an experimental setup is made in our department. The Moodle server version of 2.2.1 is installed in server hardware of Quad Core CPU with i5 processor. It has 16 GB RAM, 500 GB Hard disk and Debian 6.x as operating system. The Virtual Programming Lab (VPL)[1] is installed along with jail-server[2] in an Ubuntu-based Operating System. In addition to that, Hackystat[3] sensor server component is installed along with Moodle server that collects the data from Eclipse IDE[4] through Hackystat sensor clients installed in lab desktop machines. The PSO system and Moodle are installed separately in external servers. Data collected is pre-processed and directly stored in the Hadoop Distributed File System (HDFS). The Hadoop cluster for this experiment comprises of three separate servers, where each server has the capacity of 2000 GHz processor speed, 128 GB RAM and Cent OS 5.5 as the operating system.

B. Results and Discussion

A simulated experiment was carried out for the range of 100 to 500 students, over a total of 60 problems, and a total of 30 hours to solve these problems. Learner study experiences are randomly generated corresponding to estimates of accuracy, efficiency, and quality based on the observed distribution of real data collected from students. The generated results are subjected to K-means clustering, followed by PSO-based clustering. K-means is a partioning based algorithm; it tends to solve NP-Hard problems. However the main drawback of K-Means clustering is formation of poor clusters, due to the random selection of initial centroids. The Quality of a cluster is measured by three parameters: a) Intra Cluster Distance (ICD) and b) Inter Cluster Distance (InterCD) c) Accuracy. The Intra Cluster Distance is measured as sum of distance between all data instances belonging to the cluster and Inter Cluster distance is measured as sum of distance between all

[1] http://vpl.dis.ulpgc.es/index.php

[2] http://vpl.dis.ulpgc.es/index.php/en/documentation/18-how-to-configure-the-jail-daemon

[3] http://code.google.com/p/hackystat/

[4] http://www.eclipse.org/

the centroids. The Intra Cluster distance should be minimum and Inter Cluster distance should be maximum. The Intra Cluster Distance is lower in PSO-based clustering and it reaches the convergence value in a steady manner compared to K-Means algorithm. Moreover, the Inter Cluster Distance is consistently higher in PSO-based clustering than K-Means clustering approach. The comparison of Intra Cluster Distance is shown in Figure 2 and the Inter Cluster Distance is shown in Figure 3. Accuracy represents the quality of a cluster. The quality of a cluster is measured using the following values such as: True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The computed accuracy for PSO-based clustering and K-Means clustering is shown in Table 1.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

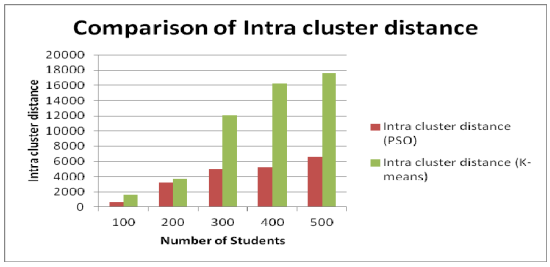

FIGURE 2: COMPARISON OF INTRA CLUSTER DISTANCE

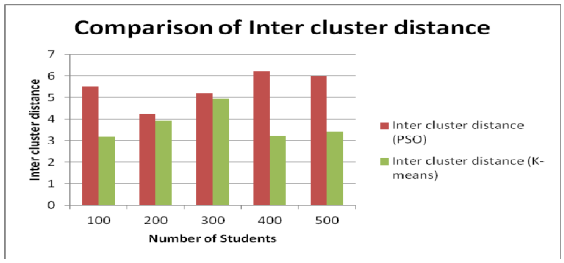

FIGURE 3: COMPARISON OF INTER CLUSTER DISTANCE

TABLE 1: COMPARISON OF ACCURACY

S.No	Accuracy in PSO-based Clustering	Accuracy in K-Means Clustering
1	0.943	0.845
2	0.912	0.81
3	0.923	0.8267
4	0.935	0.798
5	0.97	0.86

VI CONCLUSION AND FUTURE WORK

Educational institutions are ready to supplement classroom environments with online environments. Recent advances in big data learning analytics allows for the collection of continuous data from learning activities performed by learners. Such data can be used to continuously, efficiently, and accurately cluster students based on their competencies and study habits. The PSO-based approach is validated by generating simulated data based on actual/real data collected from engineering graduates. The proposed PSO-based clustering performs better than K-Means algorithm. In our efforts to scale the clustering requirements, we aim to develop a Cloud-based MapReduce framework with Hadoop environment to perform clustering in a parallel and in a distributed manner. We aim to allow students, peers and teachers to interact with the PSO-algorithm that generates additional particles for the swarm. Such an interactive approach will be experimentally tested for its accuracy of clustering in comparison with other clustering methods.

ACKNOWLEDGEMENT

The authors would like to thank and acknowledge the Indo-Canadian Shastri Institute, Canada for providing the PDIG grant to carry out this research work and for the student researcher to visit Canada from India. The authors extend their gratitude to Athabasca University, Edmonton, Canada, Anna University, Chennai, India, and NSERC, Canada for their support.

References

[1] Brusilovsky, P., Peylo, C., (2003). “Adaptive and intelligent web-based educational systems”, International Journal of Artificial Intelligence in Education, 13, 156-169.

[2] García, E., Romero, C., Ventura, S., Castro, C. (2006). “Using rules discovery for the continuous improvement of e-learning courses”, In International Conference Intellligent Data Engineering and Automated Learning. Burgos, Spain, pp. 887-895.

[3] Romero, C., Ventura, S., García E. (2008). “Data mining in course management systems: Moodle case study and tutorial”, Computers in Education, 51 (1), pp. 368-384.

[4] Romero, C., Ventura, S., & Bra, P. D. (2004). “Knowledge discovery with genetic programming for providing feedback to courseware author”, User Modeling and User-Adapted Interaction: The Journal of Personalization Research, 14(5), 425–464.

[5] Baker, R.S.J.d., Yacef, K. (2009). “The State of Educational Data Mining in 2009: A Review and Future Visions”, Journal of Educational Data Mining, 1 (1), 3-17.

[6] Geyik, K. (2007) “Clustering e-Students in a Marketing Context: A Two stage Clustering Approach”, ECEL 6th European conference on E-Learning Copenhagen Business School, pp.245-252.

[7] Valsamidis, S., Kontogiannis, S., Kazanidis, I., Theodosiou, T., & Karakos, A. (2012). “A Clustering Methodology of Web Log Data for Learning Management Systems”, Educational Technology & Society, 15 (2), 154–167.

[8] Cui X., Potok, T.E., Palathingal, P. (2005). “Document Clustering using Particle Swarm Optimization”, IEEE Swarm Intelligence Symposium, The Westin, pp. n/a.

[9] Dheeban S.G, Deepak V, Dhamodharan L, Susan Elias, (2010). “Improved personalized e-course Composition Approach using Modified Particle Swarm Optimization with Inertia coefficient”, International Journal of Computer Applications volume 1 – No.6, 102 – 107.

[10] Sridevi. U.K, Nagaveni. N., (2011). “Semantically Enhanced Document Clustering Based on PSO Algorithm”, European Journal of Scientific Research, Vol. 57, No.3, pp. 485-493.