2012 Seventh International Conference on P2P, Parallel, Grid, Cloud and Internet Computing

Using of Machine Learning into Cloud Environment (A Survey)

Managing and Scheduling of Resources in Cloud Systems

Elham Hormozi
Computer Engineering and Information Technology
Mazandaran University of Science and Technology
Babol, Mazandaran (North of Iran), IRAN
e.hormozi@ustmb.ac.ir

Hadi Hormozi
Computer Engineering and Information Technology
Islamic Azad University of Arak
Arak, IRAN
hadyhormozy@gmail.com

Mohammad Kazem Akbari
Computer Engineering and Information Technology
Amirkabir University of Technology (Tehran Polytechnic)
Tehran, IRAN
akbarif@aut.ac.ir

Morteza Sargolzai Javan
Computer Engineering and Information Technology
Amirkabir University of Technology (Tehran Polytechnic)
Tehran, IRAN
msjavan@aut.ac.ir

***Abstract*—Cloud computing is a model for delivering information technology services in which resources are retrieved from the internet through web-based tools and applications, rather than a direct connection to a server. Many companies, such as Amazon, Google, Microsoft and so on, are developing cloud computing systems and enhancing their services to provide for a larger amount of users. This technology holds a vast scope of using the various aspects of machine learning for increased performance and solving some of the challenges in front of the research community. In this survey, we investigate the effects using the concepts of machine learning on cloud environments, e.g. automated resource allocation mechanism, intelligently managing and allocating resources with SmartSLA, resources scheduling, etc.**

Keywords-cloud computing; machine learning; performance; intelligenty managing; allocating resource;

I. INTRODUCTION

Cloud computing is a recent paradigm shift from the client-server architecture which replaced mainframe computers in early 1980s. Cloud computing is a computing model, not a technology. In this model "customers" plug into the "cloud" to access IT resources which are priced and provided "on-demand". Essentially, IT resources are rented and shared among multiple tenants much as office space, apartments, or storage spaces are used by tenants. Delivered over an internet connection, the "cloud" replaces the company data center or servers providing the same service.

Thus, cloud computing is simply IT services sold and delivered over the Internet. For sustaining this shift, embedding intelligence in the cloud is a necessary requirement. Furthermore, cloud availability on the web to a large number of users calls for its ability to scale largely and provide intelligent service to the customers having varied requirements. This intelligence can allow scalability of cloud resources, enhance its performance and give its users- the cloud clients as well as back end operators- the cloud vendors and partners, a better experience using the computing paradigm. Along these lines, the possibility and the need of using machine learning concepts in cloud computing has been acknowledged by the research communities [1, 2].

In this survey, we intend to investigate managing and scheduling resources techniques in cloud computing systems. To do this, we survey some of machine learning methods such as "SmartSLA" a cost aware resource management system. SmartSLA provides intelligent service differentiation according to factors such as variable workloads, SLA levels, resource costs, and deliver improved profit margins. The system modeling module uses machine learning techniques to learn a model that describes the potential profit margins for each client under different resource allocations.

"SysWeka" which extends Weka capabilities (Weka is a data mining and machine learning tools written in Java that involves API interface and easy extensibility.) and provide a software interface for usage by higher application for managing resources on cloud systems.

Also, "Flextic" is a new job execution environment that exploits scalable static scheduling techniques. Thomas A. Henzinger and et al, evaluate Flextic on top of the Amazon EC2 cloud. Also they choose jobs from different domains: gene sequencing, population genetics, machine learning, and image processing. As an example,The *machine learning* job treats the problem of object localization in a natural image. They created a MapReduce job, where a mapper analyzes the localization for one particular image and returns the four coordinates in text form. The reducer concatenates the output of the mappers and puts it in the data store. For all jobs, they asked the user the maximum running time for each task in the job.

The remainder of this paper will be organized as follow: Automated resource allocation mechanism added to clouds systems, which will be presented in Section II. While in Section III, SmartSLA is introduced for intelligently manage the virtual resources into cloud database systems. Also, in Section IV, SysWeka will be presented and in

978-0-7695-4841-8/12 $26.00 © 2012 IEEE
DOI 10.1109/3PGCIC.2012.69

Section V, Flextic will be introduced for resource management in cloud environment. Finally, in Section VI, we will evaluat these methods, and conclude in the end of paper.

II. Automated Resource Allocation Mechanism in Clouds

Scaling of resources by a human operator is easy, but does not seem to be a good option since the size of the clouds is increasing and cloud is also including multiple web services sharing same infrastructure and even the same data [2,3].

Hence, automation of mechanism to allocate resources becomes a necessity. This automation of resource allocator for the web service needs to take into account performance history, performance problems, SLA and resource conservation issues while scaling up or down. Such a system can rely on machine learning techniques to efficiently decide the amount of resources necessary for the service. Further towards automating the cloud is the automation of user troubleshooting. Which needs to solve the issues being faced by users considering the performance delivered to the customer in past. This provides scope of utilizing concepts and ideas of machine learning for taking automation to another level. Further, machine learning can be used to monitor usage patterns and draw appropriate conclusions from them to be used in pertinent situations. Examining user behavior to understand user requirements and expectations is another aspect which can leverage on learning algorithms for better customer experience.

Navendu Jain et al. in [4] introduce an online learning approach for resource allocation to address tradeoff between computation cost and performance. Their approach is based on incorporate the choice between spot vs on-demand instances. Used algorithm dynamically adapts the resource allocation policy by learning from its performance on prior job executions, while incorporating history of spot prices and workload characteristics. In fact, they provide theoretical bounds on the performance of their algorithm, and prove that the average regret (compared to the best policy in hindsight) vanishes to zero with time. The regret of an algorithm is defined as the difference between the cumulative performance of the sequence of its decisions and the cumulative performance of the best fixed decision in hindsight.

Evaluation on traces from a large batch computing cluster shows that their algorithm outperforms greedy allocation heuristics and quickly converges to a small set of best performing policies while using a moderate number of training data samples. In addition, their method enables interpreting the allocation strategy of these policies, and allowing users to adjust them to their specific requirements. Overall, their suggest that online learning may prove to be an effective framework for adaptive resource allocation in cloud computing.

Xavier Dutreilh et al. in [12] use Reinforcement Learning (RL) for autonomic resource allocation in clouds. RL algorithms require care and expertise to deal with the main requirements of self-adapting cloud infrastructures: good allocation policies from the start prompt convergence to the optimal policy and capability to deal with evolution in the performance model of applications. This approach is particularly Well-Suited to cloud computing as they don't require a priori knowledge of the application performance model, but rather learn it as the application runs. RL faces a lot of problems [10, 11], such as having good policies in the early phases of learning, time for the learning to converge to an optimal policy and coping with changes in the application performance behavior over time. These problems is solved in [12] by using appropriate initialization for the early stages, convergence speedups applied throughout the learning phases and performance model change detection.

III. Intelligently Manage the Virtual Resource in a Share Cloud Database System

In a cloud computing environment, resources are shared among different clients. Intelligently managing and allocating resources among various clients are important for system providers, whose business model relies on managing the infrastructure resources in a cost-effective manner while satisfying the client SLAs. Pengcheng Xiong et al. present SmartSLA, a cost aware resource management system [5]. SmartSLA consists of two main components: the system modeling module and the resource allocation decision module. The system modeling module uses machine learning techniques to learn a model that describes the potential profit margins for each client under different resource allocations. Based on the learned model, the resource allocation decision module dynamically adjusts the resource allocations in order to achieve the optimum profits. The performance results show that SmartSLA can successfully compute predictive models under different hardware resource allocations such as CPU and memory, as well as database specific resources, such as the number of replicas in the database systems.

The experimental results also show that SmartSLA can provide intelligent service differentiation according to factors such as variable workloads, SLA levels, resource costs, and deliver improved profit margins. SmartSLA considers many factors in cloud computing environments such as SLA cost, client workload, infrastructure cost, and action cost in a holistic way. Experimental studies on benchmark data and real-life workloads demonstrated that such an intelligent resource management system has great potentials to improving the profit margins of cloud service providers [5, 6].

IV. Resource Management on Cloud Systems by SysWeka

"SysWeka" is a middleware platform. Machine learning techniques based on systems-oriented Weka are adopted to build this platform. "SysWeka" provides a software

interface for usage by higher cloud application for managing resources on cloud systems. This framework provides a fast and easy approach to build applications based on lower ML infrastructures. Figure 1 shows the architecture of this platform.

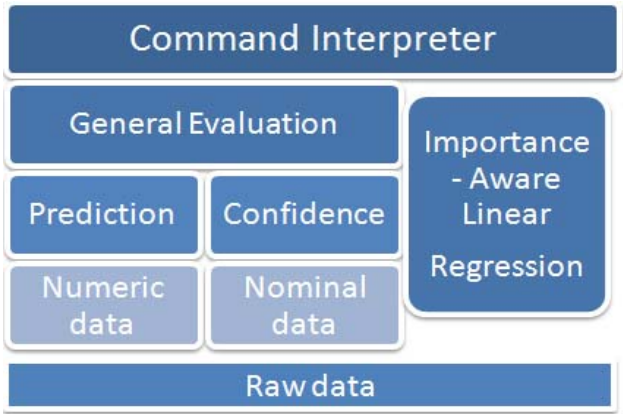

Figure 1: SysWeka platform architecture

In [7], we can understand that although linear regression has been widely used in many fields to build models with successful results, it can't produce benefit outcome. In contrast, Bayesian network acts as a more proper choice and performs much better. Moreover, confidence prediction is presented and developed to measure the accuracy of predictions, which gives us another opportunity to make a further decision whether to trust the predictive result or not. Besides, importance-aware linear regression has been proposed and derived by mathematics. Experiments evaluation also shows a different view of the importance-aware dataset, which indicates a promising application usage in the future work. Instances can be specified by different importance and thus may create more valuable results.

V. RESOURCE SCHEDULING IN CLOUDS

The price that a user is required to pay for the execution of a particular job in the cloud depends on the length of the job and the amount of data transfer involved in the job execution. This model is intuitive for long term rentals of computing instances on the cloud.

Flextic is a new job execution environment that exploits scalable static scheduling techniques to provide the user with a flexible pricing model such as a tradeoff between different degrees of execution speed and price, and on the other hand, reduce scheduling overhead for the cloud provider (The workflow of Flextic is shown in Figure 2). Also, Flextic is a new approach to scheduling jobs on the cloud systems by separating the concerns of the end users and the cloud provider and bridging the gap between the pricing needs of the two sides[8]. Leaving the responsibility of scheduling the jobs with the cloud provider enables the provider to achieve good utilization of its resources.

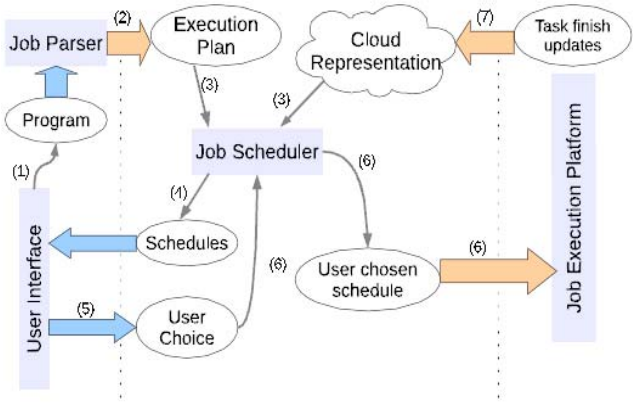

Figure 2: Flextic Workflow

Henzinger and et al. in [9] evaluated Flextic on top of the Amazon EC2 cloud. They choose jobs from different domains: gene sequencing, population genetics, machine learning, and image processing evaluated the scheduling of Flextic. According to their experiment, Flextic has a low scheduling latency. Moreover, static scheduling is more user-friendly, as the precomputed schedule allows to quote a price for the computation. To implement Flextic, they need to tackle some challenges in scheduling. First of all, they need static schedulers that can efficiently schedule large jobs on large data centers. Recently, they developed exciting static scheduling techniques for large jobs on clouds, based on ideas of abstraction refinement (AR) [9, 10].

Indeed, leaving the responsibility of scheduling the jobs with the cloud provider enables the provider to achieve good utilization of its resources [10]. For computing the scheduling latency, they build a cloud by Amazon EC2 instances consisting of small, large, and extra-large types. In total, their cloud consists of 200 virtual cores.

Table 1 shows the time required by Flextic to obtain ten different schedules. They also give the size of the execution plan, and the time required to create it after Job EP Details Scheduling nodes edges time latency Gene Sequencing fetching the required information from Amazon S3. They observe that for the machine learning job with around 6700 tasks, Flextic requires only 2.3 seconds to compute around ten different schedules. The time required by the static scheduler depends on the regularity of the job and the quality measures set for scheduling. They set the quality measure as 90% cloud utilization.

Job	EP Details			Scheduling
	nodes	edges	time	latency
Gene Sequencing	11	22	0.026 s	0.01 s
	21	42	0.043 s	0.02 s
Machine Learning	183	550	0.184 s	0.26 s
	6711	7732	1.251 s	2.29 s
Population Genetics	22	45	0.063 s	0.02 s
	210	421	0.195 s	0.31 s
Image Processing	401	802	0.263 s	0.43 s
	2005	2406	0.731 s	1.36 s

Table 1: Evaluation of the time required for static scheduling for two examples of execution for each job.

Josep Ll. et al. in [13], by using machine learning presented adaptive scheduling on power-aware managed data-centers. Nowadays, optimizing the management of data-centers to make them efficient, not only in economic values; nevertheless, for power consumption, the automation of several systems such as job scheduling and resource management is required. They propose an autonomic scheduling of tasks and web-services over cloud environments. Actually, they focused on the profit optimization by executing a set of tasks according to service level agreements minus its costs like power consumption. The data mining and machine learning techniques are in charge of obtaining the characteristic functions relating the amount of load with the required resources. They shown that it is possible to model jobs and system behaviors towards resources and SLA metrics in an automatic manner through machine learning. Also they apply them to full-scale data-center models, letting schedulers and decision makers to have more adjusted estimation functions a priori, in order to make better their decisions for each system and kind of job. The main idea is that a mathematical model of the system, involving the function of revenue, the function of power versus resource usage, the function of SLA fulfillment and the learned function of response time vs given resources and system status; fulfilled with the system parameters, the kind of jobs and loads, and the predicted expected minimum resource usages, can be processed by an exact solver, or approximated algorithms solving in less computational time and space with a tolerable approximate solution. On the other hand, VMware, the global leader in cloud infrastructure, delivers customer-proven virtualization solutions that significantly reduce IT complexity. VMware accelerates an organization's transition to cloud computing, while preserving existing IT investments and enabling more efficient, agile service delivery without compromising control. The VMware companies are investigating how machine learning can improve virtualization technology on the cloud. They will involve developing tools and techniques to use machine learning for scheduling and resource management within a single ESX host server, a data center as well as across a distributed cloud.

VI. Dynamic and Flexible Resource Sharing on the Cloud

Unfortunately, sharing a cluster efficiently between two or more frameworks is difficult. Many operators statically partition their clusters at physical machine granularities, making poor overall resource utilization. To reduce the challenges of building distributed applications, researchers have developed a diverse array of new software frameworks for clusters. Static partitioning makes it expensive to share big datasets between 2 computing frameworks such as Hadoop and MPI. One must either copy the data into a distinct cluster for each framework, consuming extra storage, or have the frameworks read it across the network, reducing performance [16]. Apache Mesos is a cluster manager that provides efficient resource isolation and sharing across distributed applications, or frameworks. It can run Hadoop, MPI, Hypertable, Spark (a new framework for low-latency interactive and iterative jobs), and other applications. Mesos is open source in the Apache Incubator. Hindman and et al. in [16] developed Spark, a framework for iterative applications and interactive data mining that provides primitives for in-memory cluster computing. Spark is suitable for machine learning and graph applications, and for interactive data mining, where a user can load a dataset into memory it repeatedly.

VII. Evaluaton of Methods

Pengcheng and et al. in [5], use the default CPU/Memory allocations (50 shares/512MB) to both of the clients as the baseline case. They fix the number of replicas as 2, i.e. M (k) = 2. For the overall performance, the total weighted SLA penalty cost is 2802 for the baseline and 2364 for Smart-SLA. That is, SmartSLA reduces about 15% SLA penalty cost by dynamically tuning CPU and memory shares between gold and silver clients. They analyze the cost model by hanging the number of database replicas and show that by taking this cost model into consideration; SmartSLA can further improve the cost efficiency. Also, they used machine learning techniques to learn a system performance model through a data-driven approach. The model explicitly captures relationships between the systems resources and database performance. Based on the learned predictive model, they designed an intelligent resource management system, SmartSLA. SmartSLA considers many factors in cloud computing environments such as SLA cost, client workload, infrastructure cost, and action cost in a holistic way. SmartSLA achieves optimal resource allocation in a dynamic and intelligent fashion. Experimental studies on benchmark data and real-life workloads demonstrated that such an intelligent resource management system has great potentials in improving the profit margins of cloud service providers. Typically, SLAs will guarantee most aspects of service delivery, including both technology aspects and customer service aspects. The customer service guarantees often include availability of support resources and response

time on technical support requests. The technology guarantees can include error resolution time guarantees, system response time guarantees, and almost always include system availability or uptime guarantees [14].

Zhenyu and et al. in [7] show some important machine learning techniques and present SysWeka platform and its performance evaluation. From this SysWeka platform, they can conclude that although linear regression has been widely used in many fields to build models with successful results, it cannot produce benefit outcome in our scenarios. In contrast, Bayesian network acts as a more proper choice and performs much better. Moreover, confidence prediction is presented and developed to measure the accuracy of predictions, which gives us another opportunity to make a further decision whether to trust the predictive result or not. Besides, importance-aware linear regression has been proposed and derived by mathematics. Experiments evaluation also shows a different view of the importance-aware dataset, which indicates a promising application usage in the future work. Instances can be specified by different importance and thus may create more valuable results in further studies. Bayesian network using *K2* as search algorithm and Simple Estimator as estimator with max three parents per node performs advantageous classification with three different kinds of testing methods-use training set, cross-validation and percentage split. Though the graphical model is more complex than other models, this method does provide high performance within reasonable time.

Thomas and et al. in [9], evaluate the scheduling of Flextic. They show that Flextic has a low scheduling latency. For example, on a cloud with 200 cores, for a MapReduce job with around 6500 tasks, Flextic can compute ten different schedules in around two seconds. Also, they consider the image processing MapReduce job, and compare the performance of Flextic with a Hadoop scheduler on Amazon EC2. They observe that due to the large communication overhead for Hadoop at runtime, Flextic outperforms Hadoop by up to 15% in job execution time. There is the risk that the transaction will not be completed and the exchange will be lost or possibly worse, the wrong investment decision will be made with even more costly results.

Investors are freed from the time constraints of the 1031 Exchange rules by completing an exchange into and out of a Flextic property, providing time to evaluate investment goals and solutions on the investor's timetable. Flextic coordinates the exit of investors from it by repurchasing Flextic interests from investors which, in turn, makes those interests available to more investors. Thus, there is assured a constant supply of tenant-in-common interests of flexible investment sizes to accommodate most investors' needs.

Table 2: Evaluation of Methods

Method	Year	Metric	Evaluation
SmartSLA [5,14]	2010	SLA penalty cost	Reduces 15% SLA penalty cost
		Error resolution time	guaranteed
		System response time	guaranteed
		Availability	guaranteed
Sys Weka [7]	2010	Accuracy	%98 Accuracy
		High performance	Bayesian network
Flextic [8]	2011	Scheduling	Low scheduling overhead, reduces job duration
		Time	To evaluate investment solutions on the timetable
		Availability	Increasing availability

VIII. CONCLUSIONS

In this paper we investigated using of machine learning into cloud systems. Automation of resource management and scheduling in cloud environments requires knowledge about the system to act in an "intelligent" way. As shown here machine learning can provide this knowledge and intelligence, as well as adaptivity. Efficient management of resources at cloud scale while providing proper performance isolation, higher consolidation and elastic use of underlying hardware resources is an importnant key to a successful cloud deployment. Also, leaving the responsibility of scheduling the jobs with the cloud provider enables the provider to achieve good utilization of cloud resources. We hope that our survey will help motivate future research in this critical area to solve practical issues.

ACKNOWLEDGMENT

The authors would like to thank all those who contributed to this paper. Further to this, we gratefully acknowledge those in the Cloud Computing Research Center at the Computer Engineering and Information Technology department, AmirKabir University, IRAN.

[1] David Burford, "Cloud computing: a brief introduction,", LAD NTERPRIZES, February 20, 2010

[2] Gurmeet Singh, "Scope of machine learning in cloud computing,", Under the guidance of Dr. Diganta Goswami, 8-11-2010.

[3] R. Geambasu, S. D. Gribble, and H. M. Levy., "CloudViews: communal data sharing in public clouds". In HotCloud 09 Workshop on Hot Topics in Cloud Computing, 2009.

[4] Navendu Jain, Ishai Menache, Ohad Shamir, "On-demand or Spot? learning-Based resource allocation for delay-tolerant batch computing,", Microsoft Research, 2011.

[5] Pengcheng Xiong, Yun Chi, Shenghuo Zhu, Hyun Jin Moon, Calton Pu, Hakan Hacıg˘um¨us¸, “Intelligent management of virtualized resources for database systems in cloud environment,”, 2010.

[6] D. Florescu and D. Kossmann, “Rethinking cost and performance of database systems,”, SIGMOD Rec., vol. 38, pp. 43–48, June 2009.

[7] Zhenyu Fang,”Resource management on cloud systems with machine learning”, Master thesis, Master in Information Technology arcelona School of Informatics Technical University of Catalonia, July, 2010.

[8] Thomas A. Henzinger, Anmol V. Singh, Vasu Singh, Thomas Wies, Damien Zufferey IST Austria, “Static scheduling in clouds”,2011.

[9] T. A. Henzinger,V. Singh, T.Wies, and D. Zufferey. Scheduling large jobs by abstraction refinement. In EuroSYS, pages 329–342, 2011.

[10] G. Tesauro, N. K. Jong, R. Das, and M. N. Bennani, “A hybrid reinforcement learning approach to autonomic resource allocation”, in Proc. Of the 2006 IEEE Conf. on Autonomic Computing (ICAC). IEEE Computer Society, 2006, pp. 65–73.

[11] J/Rao, X. Bu, C. -Z. Xu, L. Wang, and G. Yin, “Vconf: a reinforcement learning approach to virtual machines auto-configuration”, in Proc. Of the 6th Int. conf. on Autonomic computing (ICAC), 2009, pp. 137–146. P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” Tech. Rep., July 2009. [Online]. Available: http://www.csrc.nist.gov/groups/SNS/cloud-computing/ [retrieved: February 15, 2012].

[12] Xavier Dutreilhy, Sergey Kirgizov, Olga Melekhova, Jacques Malenfant, Nicolas Rivierrey and Isis Truck, “Using reinforcement learning for autonomic resource allocation in clouds: towards a fully automated workflow”, Paris, 75005, France, 2011.

[13] Josep Ll. Berral, Ricard Gavald`a, Jordi Torres, “Adaptive scheduling on power-aware managed data-centers using machine learning”, Universitat Polit`ecnica de Catalunya and Barcelona Supercomputing Center, 2011.

[14] Chris K.”Smart SLA design for your SaaS” April 7th, 2010. http://blog.inetu.net/2010/04/smart-sla-design-for-your-saas/, [retrieved: July, 2011].

[15] Richard T.B. Ma, Dah Ming Chiu, John C.S. Lui, Vishal Misra and Dan Rubenstein, “On resource management for cloud users: a generalized kelly mechanism approach”, Electrical Engineering, May, 2010.

[16] B ENJAMI N H INDMAN, ANDY KONWINS K I, ”Mesos, flexible resource sharing for the cloud”, August 2011.