http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (23): 7275-7279, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.7275.7279

Toward Green Cloud Computing: An Attribute Clustering Based Collaborative Filtering Method for Virtual Machine Migration

[1,2]Zhang Liu-Mei, [1]Ma Jian-Feng, [1]Wang Yi-Chuan and [1]Lu Di
[1]School of Computer Science and Technology, Xidian University, 710071, Xi'an, China
[2]School of Computer Science, Xi'an Shiyou University, 710065, Xi'an, China

Abstract: In this study, an attribute clustering based collaborative filtering algorithm is depicted for virtual machine migration towards green Cloud computing. The algorithm utilizes similarity characteristics of virtual machine task related attributes, especially CPU related attributes, to filter redundant data by feature selection. Then by referencing K-Means clustering to effectively solve the rating scale problems existing in the traditional collaborative filtering recommendation algorithm. Experiments use virtual machine task related information for clustering the data. By integration of a scaled rating scheme on task related properties and the collaborative filtering philosophy to provide migration recommendation for system administrators.

Key words: Green cloud computing, VM migration, collaborative filtering

INTRODUCTION

Cloud computing maximizes the resource usage of cloud servers then traditional computing turns to be green. But in recent years, as Cloud computing services expand in a high speed, number ofenvironmentalfactors has been considered according to the growing number of Cloud server clusters.

In green computing, numerous researchers have paid much attention on the way of reducing the energy consumption. Strategies of green energy saving for cloud computing platform were prosed and based on fuzzy comprehensive evaluation, the method and model of green cloud computing platform evaluation were put forward (Gong *et al*., 2013). Moreover, framework is proposed to automatically manage resources of cloud infrastructures in order to reduce as much as possible the amount of energy used for providing services (Guazzone *et al*., 2012). One the other hand, different ideas towards green cloud computing approach has formed by considering energy efficient computing (Jain *et al*., 2013). While, for green cloud data centers, dynamic data aggregation algorithm is proposed (Xu *et al*., 2012). But whether applying new architectures or inventing new energy efficient hardware are naturally go against with green Cloud computing. That is to say, recycle the retired architecture or hardware also consumes large energy. Therefore, to enhance green Cloud computing on current infrastructure is of economic interest.

As previous mentioned factors mainly related to the growing numbers of Virtual Machine (VM) instances, to study the behavior of VM instances is to some extent of importance of developing new techniques of green Cloud computing. If behavior of VM instances can be measured, to classify similar behaved instances is a meaningful research topic that may further imply the rational migration of VM instances. Then, we are able to migrate frequently used instances and left over often idled instances. Then idle instances resided cloud servers are able to be switched off the power. Simply because the best way to limit the power consumption of the cloud server is to shut off the power when VM instances running on is not in use.

As Cloud computing can enable more energy-efficient use of computing power, especially when the computing tasks are of low intensity or infrequent (Baliga *et al*., 2011). We assume that computing tasks are the criteria for describing VM instance behavior. Because information from a task can be used for many purposes, such as personnel selection and training, tool or equipment design, procedure design (e.g., design of checklists or decision support systems) and automation (Hackos and Redish, 1998).

Collaborative filtering has been used in board areato generate recommendations. Such as Zhang *et al*. (2010) proposed a spatial clustering-based collaborative filtering algorithm which separates the procedure of recommendation into offline and online phases. Cheinshung Hwang (2006) proposed a cluster-based collaborative filtering algorithm based on the fuzzy set theory to predict Web pages. In this study, we assume the users as VM instances subscribers. Then

Corresponding Author: Liumei Zhang, School of Computer Science and Technology, Xidian University, 710071, Xi'an, China

similar instances is able to be grouped according to their subscriber activities which may defined by computing tasks. Each attribute data of computing task then is the source of the metrics to measure the instance activity. Stable instances are the instances with higher usage which need to be migrated. While unstable instance are the instances that with lower usage which need to reside in the physical machine to be switched off. Thus, creating a rating table to map raw data to those metrics is significant for applying collaborative filtering. Hitherto, clustering can allocate instances into same cluster according to the instances similarity. Therefore, we are able to predict the instance rating by analyzing the cluster commonalities. In this study, we only focus on CPU related data to propose an attribute clustering based collaborative filtering algorithm. Such algorithm may produce clusters with similar instance rating. Then it finds the closest cluster for the new instance based on their rating scores. Finally, the algorithm generates recommendation for the new instances by referencing the instances raw task data.

ATTRIBUTE CLUSTERING BASED COLLABORATIVE FILTERING

For virtual machines running in the cloud cluster, if the average CPU rate is low, then the running cloud service consumption of CPU resources is low and even often dormant. Then migrate this virtual machine to another physical machine won't take up too much of the CPU resource. Therefore, migration of such virtual machine can save electric power effectively. Similar manner also applied to the virtual machine that has low maximum CPU rate.

Now we analysis the CPU instruction cycle that VM task occupied. If the CPU task period is short that indicate the CPU of virtual machine has frequent task scheduling. Otherwise, it is infrequent and is able to be migrated. Cycles per instruction describes the type characteristics of instruction that CPU executed. Short cycles per instruction indicates the instruction is the conventional instruction such as operation instruction, short instruction and short data operation instruction. Moreover, those CPU instruction is used only call register such as EAX etc.Therefore, the virtual machine can be migrated to other physical machine for long-term operation. Long cycles per instruction indicates the instruction requires co-processor involved in operation, big data operation or conduct abnormal control etc.The operation duration is not long and even soon the virtual machine will be closed.So it is not necessary to migrate.

Thus, we give the definition of Alternative Cluster of Instance Migration.

Definition: A cluster C is defined as the Alternative Cluster of Instance Migration (ACIM), if C with instances that has task usage data that composed with high CPU rate, maximum CPU rate, cycles per instruction and long task duration.

According to the definition of ACIM, we further discuss which virtual machine should first be migrated in ACIM.

Instance attributes similarity selection: As similarity measure for clusteringwas sometimes confused with the similarity measure for collaborative filtering (Jia *et al*., 2010). In this study, for clustering convenience, we introduce an instance attribute space to summarize the instance ratescore into number of feature clusters which denoted as $\Omega = A_1, A_2, ..., A_k$. Where k is the number of attributes of an instance. For some instances, however, a specific attribute may have multiple attribute values.

Therefore, this study adopts the single attribute to obtain similarity of an instance on a particular attribute. Thensum all the similarityof attributes and then calculatethe average similarity, the similarity between instances. For example, attributes space of the instance I_1 and I_2 is denoted as $A1 = \{a_{11}, a_{12}, \ldots, a_{1k}\}$. Therefore, we can get the instance attribute matrix $n \rightarrow k$ from attribute A1 as shown in Table 1.

In Table 1, column k indicates attribute A1 has k values while row n indicates the number of instances. 1 and 0 represents whether A_1 attribute of the instance conform to the attribute values of all other instances. 1 is conform to, 0 is not conform to.

The sample space then is reduced, now we need to choose from which a virtual machine needs to be migrated. Afterinstance attribute matrix is constructed, we are able to calculate the similarity for further measurement of similarity degree between instance I_1 and I_2 toward attribute A1. A set of feature vector is then formed with respect to the conformity value of an instance towards certain attribute. For example, the feature vector of user I_1 and I_2 toward attribute A_1 is depicted as $\overrightarrow{I_1A_1} = \{i_1a_{11}, i_1a_{12}, \ldots, i_1a_{1k}\}$ and $\overrightarrow{I_2A_1} = \{i_2a_{11}, i_2a_{12}, \ldots, i_2a_{1k}\}$.

Table 1:Instance attribute matrix

	a_{11}	a_{12}	…	a_{1i}	…	a_{1k}
I_1	1	0	…	1	…	0
I_2	0	1	…	0	…	1
…	…	…	…	…	…	…
I_i	1	0	…	1	…	0
…	…	…	…	…	…	…
I_n	0	1	…	0	…	1

Then the attribute similarity of I_1 and I_2 toward attribute A_1 can be represented as:

$$S_1 = \text{sim}(I_1A_1, I_2A_1) = 1 - \frac{\overrightarrow{I_1A_1} \oplus \overrightarrow{I_2A_1}}{K} = 1 - \frac{\sum_{i=1}^{k} \overrightarrow{I_1a_{1i}} \oplus \overrightarrow{I_2a_{1i}}}{K} \quad (1)$$

where, $\text{sim}(I_1A_1, I_2A_1)$ is the similarity, $\overrightarrow{I_1A_1} \oplus \overrightarrow{I_2A_1}$ is the attribute value with no commonality between I_1 and I_2 toward A_1. Such attribute value is capable for the generation of probability value of certain attribute of the instance after XOR operation. Then the attribute value, with no commonality, is summed and divided by K. The result depicts the incoherence degree towards A_1. Note K is the total value numbers of A_1.

Next, the average trust degree of similarity, EA(sim), between I_1 and I_2 is calculated as:

$$\overline{S} = EA[\text{sim}(I_1, I_2)] = \frac{\sum_{i=1}^{m} (I_1A_i, I_2A_i)}{K} \quad (2)$$

Then, the expectation value of trust degree among all attributes is obtained to describe the mean value of similarity between instances. m is the number of attributes used for such description.

Therefore, similarity between instances can be obtained:

$$S = \text{sim}(I_1, I_2) = \frac{\sum_{i=1}^{k} (S_1 - \overline{S})}{\sqrt{\sum_{i=1}^{k} (S_1 - \overline{S})^2}} \quad (3)$$

It describes the similarity between instance I_1 and I_2. Where k is the total value numbers of A_1.

Instance similarity clustering: In this study, we simply use the k-means algorithm for the instancesgroups' analysis. Assuming dataset and cluster centroid is n dimensional vector, we repeat following two steps until the convergence:

Step 1: For each x^i, obtain the nearestcentroid j and then mark it into different categories

We need to assign x^i to cluster j for assign all points into its nearest centroid:

$$\text{set } c(i) = \arg\min \|x_i - u^j\| \quad (4)$$

Step 2: Updates the cluster centroid to the average value of all points, to determine the new centroid:

$$\text{set distance} = d$$
$$\text{set clster\_center} = EA\text{ (distance)} \quad (5)$$

Assuming there are n instances, thus the recommended collection is $I = \{I_1, I_2, \ldots, I_n\}$. After treated with K - means algorithm, the clusters can be described as $C = \{c_1, c_2, \ldots, c_j\}$. Where j is the total cluster numbers, c_i contains the instances with high preference and interest similarity. The realization shows as follows:

Input: ClusterNum j and Matrix(n×k);
Output: No. of cluster about matrix is j;

Step 1: Searching n instances within instance attribute matrix, depicted with collection $I = \{I_1, I_2, \ldots, I_n\}$
Step 2: Randomly choosing j instances. Setting their attribute data as the initial cluster centroid, depicted with collection $C' = \{c'_1, c'_2, \ldots, c'_j\}$;
Step 3: Empty j clusters, depicted with collection $C = \{c_1, c_2, \ldots, c_j\}$
Step 4: Perform the following actions on the rest of the instances:

Instances-clustering-algorithm

- For all $i_i \in I$ do
- For all $c' \in C'$ do
- sim (ci, c'i)
- End for
- sim (ii, c'm) = max { sim (ii, c'm), …, sim (ii, c'm)}
- $c'_m = c'_m \cup i_i$
- End for

where, in the Instances-clustering algorithm, sim (ii, c'i), is the similarity between i_i and centroid c'_i, sim (i_i, c'_m) = max {sim (i_i, c'_m), …, sim (i_i, c'_m)} and $c'_m = c'_m \cup i_i$ is the clustering process.

Step 5: Calculate the mean value of all instancess in the new cluster and update the centroid
Step 6: Repeat step 4-5 until centroid is stable, output s clusters

With Instances-clustering-algorithm, we can find the VM instances that have highest similarity. For VM instances in ACIM, if one VM instance has highest similarity with the to be migrated VM instance cluster, it also should be migrated when considering CPU usage and CPU cycle factors for task. Thus we get lemma 1.

Lemma 1: For virtual machine inACIM, the virtual machine with highest similarity should first be migrated.

EXPERIMENT

Experimental data is from Google clusterdata-2011-1 dataset (Reiss *et al*., 2011) which provides data from a 12k-machine cell over about a month-long period in May 2011.

Based on taskusagetable part-00000-of-00500, by using Loop Attributes operator in Rapidminer with parameter configured as 500 runs on attribute machine ID, we have aggregated 10218 rows of the raw data. That forms our task attribute matrix which is a 10218.16 matrix. In this experiment, we have selected 4attributes for representing our design. They are CPU rate, maximum CPU rate, task duration and cycles per instruction. Then 10218 data were pivoted and reconstructed according to

Table 2: Attributes partition scheme

Attributes	Partition_1	Partition_2	Partition_3	Partition_4	Partition_5
CPU rate	0.023	0.207	0.101	0.002	0.053
Maximum CPU rate	0.005	0.069	0.230	0.129	0.414
Cycles per instruction	15.483	3.548	1.105	7.109	25.368
Task duration	314871.340	542366108.790	299235294.110		

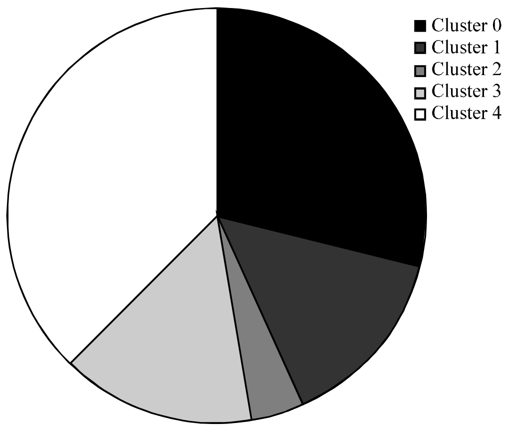

Fig. 1: Final rating cluster

the machine ID. The task attribute matrix is transformed to a 500 n matrix. Beside this, attribute data are partitioned according to their real value, respectively for further generate the instance attribute matrix. Table 2 indicates the partition scheme with centroid within each attribute, respectively. Typically, a VM instance is the stable instance once its task usage shows information with low value of CPU rate, maximum CPU rate, cycles per instruction and high value of task duration.

With this instruction, we determined the instance attribute matrix by assigning 1 to the data belongs to partition_4 in attribute cluster CPU rate; partition_1 in attribute cluster maximum CPU rate; partition_3 in attribute cluster cycles per instruction; partition_3 in attribute cluster task duration. The rest of the data are assigned with 0. Finally, 4 instance attribute matrix, each is a 500 10 matrix, are formed as 10 columns of the dataset is selected.

For each machine ID, we then sum its a_{11}, a_{12}, ..., a_{1k}, then A_1, A_2, A_3, A_4 to get the final rating.

Next, we useK-means to cluster the final rating data of all machine IDs. Figure 1 shows the result.

From Table 3 we can summarize that Cluster 3 received highest value of rating scores. According to the lemma 1 cluster 3 is become the candidate setofmigration. Then we are able to assume that the VM instances contained in this cluster are the instances which need to be migrated.

Next, we calculate the similarity of instances that outside the 500 instances sample. To demonstrate our design, we have selected 5 other machine IDrelated data. Then, we assume these 5 instances are the instances reside on a same physical machine. Again, 4 attributes and 50 records are selected for pivoting a 5×10 matrix. Attribute data are grouped according to Table 2. 0 and 1 are assigned to this matrix. According to Eq. 1-3, similarity between each instance in the new

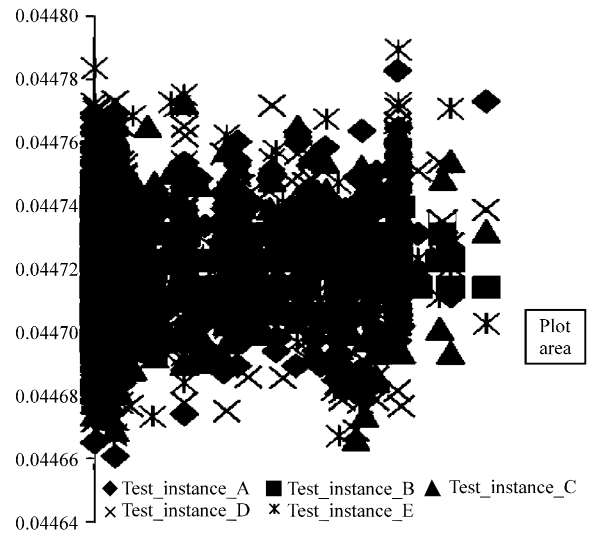

Fig. 2: Scatter plot of similarity of test instances

Table 3:Final rating cluster centroid

Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
18.683	14.778	10.421	27.671	22.862

Table 4:Cluster identification result of test instances

Cluster assigned	MachineID	Similarity
Test_instance_A		
Cluster 4	4820022347	0.044783034
Test_instance_B		
Cluster 1	257501970	0.04474831
Test_instance_C		
Cluster 0	1429206364	0.044772832
Test_instance_D		
Cluster 3	336069364	0.044773205
Test_instance_E		
Cluster 4	4820250647	0.044789675

From Fig. 2 we can get the similarity plot between the selected test instances and 500 instances. While Table 4. show us the highest value of similarity and it's correspondent Machine ID of all test instances. Obviously, Test_instance_D has highest similarity with instance with Machine ID336069364 which belongs to cluster 3. That is to say, from all the test instances Test_instance_Dis the one that needs to migrate to a stable physical machine that always powered on. Once after Test_instance_Dfinishing the migration, we are able to switch off the physical machine that it originallyresides on.We simply assume this physical machine is an IBM System x3850 X5(7145N09)which has 1.975 kw power supply. Then it is able to save much electric power after the migration.

CONCLUSION

In this study, an attribute clustering based collaborative filtering method has been proposed for identifying virtual machine instances which suitable for migration. The method first introduced a VM instance attribute space to summarize the instancerate scoreinto number of feature clusters. Then it calculates similarities between virtual machine instances. The method is verified with real dataset and thus achieved acceptable result to identify the stable instances that need to be migrated. Power off the physical machine which with only unstable instances resides on definitely decreases the energy consumption of the cloud servers. Therefore, it is practical for Green Cloud computing.

ACKNOWLEDGMENT

This study was supported in part by Program for Changjiang Scholars and Innovative Research Team in University No. IRT1078; The Key Program of NSFC-Guangdong Union Foundation No. U1135002; The Fundamental Research Funds for the Central Universities No. JY0900120301.

REFERENCES

Baliga, J., R.W.A. Ayre, K. Hinton and R.S. Tucker, 2011. Green cloud computing: Balancing energy in processing, storage and transport. Proc. IEEE, 99: 149-167.

Gong, L., J. Xie, X. Li and B. Deng, 2013. Study on energy saving strategy and evaluation method of green cloud computing system. Proceedings of the 8th IEEE Conference on Industrial Electronics and Applications, June 19-21, 2013, Melbourne, Australia, pp: 483-488.

Guazzone, M., C. Anglano and M. Canonico, 2012. Exploiting VM migration for the automated power and performance management of green cloud computing systems. Proceedings of the 1st International Workshop on Energy Efficient Data Centers, May 8, 2012, Madrid, Spain, pp: 81-92.

Hackos, J.T. and J. Redish, 1998. User and Task Analysis for Interface Design. John Wiley and Sons, New York, USA., ISBN-13: 9780471178316, Pages: 488.

Hwang, C.S., 2006. Integrating fuzzy partitional clustering and collaborative filtering for web page prediction. WSEAS Trans. Inform. Sci. Appl., 3: 2094-2099.

Jain, A., M. Mishra, S.K. Peddoju and N. Jain, 2013. Energy efficient computing-green cloud computing. Proceedings of the International Conference on Energy Efficient Technologies for Sustainability, April 10-12, 2013, Nagercoil, India, pp: 978-982.

Jia, R.F., M.Z. Jin and C. Liu, 2010. A new clustering method for collaborative filtering. Proceedings of the International Conference on Networking and Information Technology, June 11-12, 2010, Manila, Philippines, pp: 488-492.

Reiss, C., J. Wilkes and J. Hellerstein, 2011. Google cluster-usage traces: Format + schema. Google Inc., White Paper. http://googleclusterdata.googlecode.com/files/Google%20cluster-usage%20traces%20-%20format%20+%20schema%20(2011.10.27%20external).pdf

Xu, X.L., G. Yang, L.J. Li and R.C. Wang, 2012. Dynamic data aggregation algorithm for data centers of green cloud computing. Syst. Eng. Electron., 34: 1923-1929.

Zhang, F., H. Sun and J. Chang, 2010. A spatial clustering-based collaborative filtering algorithm in mobile environment. J. Comput. Inform. Syst., 6: 2297-2304.