Received: 24 August 2016 | Revised: 22 November 2017 | Accepted: 27 November 2017
DOI: 10.1002/widm.1241

WILEY WIREs DATA MINING AND KNOWLEDGE DISCOVERY

ADVANCED REVIEW

Mining association rules for admission control and service differentiation in e-commerce applications

James Xue[1] | Stephen Jarvis[2]

[1]Faculty of Arts, Science and Technology, University of Northampton, Northampton, UK

[2]Department of Computer Science, University of Warwick, Coventry, UK

Correspondence
James Xue, Faculty of Arts, Science and Technology, University of Northampton, Park Campus, Boughton Green Road, Northampton NN2 7AL, UK.
Email: james.xue@northampton.ac.uk

Workload demands in e-commerce applications are very dynamic in nature, therefore it is essential for internet service providers to manage server resources effectively to maximize total revenue in server overloading situations. In this paper, a data mining technique is applied to a typical e-commerce application model for identification of composite association rules that capture user navigation patterns. Two algorithms are then developed based on the derived rules for admission control, service differentiation, and priority scheduling. Our approach takes the following aspects into consideration: (a) only final purchase requests result in company revenue; (b) any other request can potentially lead to final purchase, depending upon the likelihood of the navigation sequence that starts from current request and leads to final purchase; (c) service differentiation and priority assignment are based on aggregated *confidence* and average *support* of the composite association rules. As identification of composite association rules and computation of *confidence* and *support* of the rules can be pre-computed offline, the proposed approach incurs minimum performance overheads. The evaluation results suggest that the proposed approach is effective in terms of request management for revenue maximization.

This article is categorized under:
Application Areas > Science and Technology
Algorithmic Development > Association Rules
Algorithmic Development > Web Mining

KEYWORDS
admission control, association rule, cloud computing, e-commerce, service differentiation

1 | INTRODUCTION

Workload demands in e-commerce applications are usually very bursty in nature, thus it is difficult to predict the workload level at a certain point in time. Moreover, due to special events or seasonal effects, some e-commerce sites are sometimes subject to huge increases in demand, which can result in system overloading. During an overloaded period, system response time may grow to an unacceptable level, and the exhaustion of resources may cause the service to behave erratically or even crash. In a very competitive e-commerce market, any user frustration due to long server response time will likely lead to less use of the service, switching to another competitor's service, and ultimately loss of revenue. Therefore, good resource management strategies need to be implemented to deal with server overloading situations, and to maximize company revenue while maintaining the service level agreements (SLAs).

Some internet service providers (ISPs) rely on resource provisioning techniques to deal with spikes in workload demands (Urgaonkar, Shenoy, Chandra, & Goyal, 2005; Villela, Pradhan, & Rubenstein, 2007; Xue, Chester, He, & Jarvis, 2008). The common practice is to employ both proactive and reactive provisioning techniques (Urgaonkar et al., 2005). Proactive provisioning is used to deal with long-term predictable changes in workload demands due to seasonal effects or special promotion events. Short-term fluctuations are very difficult to predict, therefore reactive provisioning is more appropriate. ISPs always aim at maximizing their profits by making the best use of their resources, that is, while maintaining the SLAs, they will try to allocate resources to as many e-commerce companies as possible. In system overloading situations, unless e-commerce companies are willing to pay for extra resources, ISPs usually apply admission control (AC) Schemes (Ashraf, Byholm, & Porres, 2012; Elnikety, Nahum, Tracey, & Zwaenepoel, 2004; Khojasteh, Misic, & Misic, 2015; Konstanteli, Cucinotta, Psychas, & Varvarigou, 2014; Tomas & Tordsson, 2013; Wu, Garg, & Buyya, 2012) to their services. By rejecting less important requests when the system is overloaded, AC schemes can guarantee the performance of internet services.

In addition to AC, services differentiation and service degradation can also be applied to deal with system overloading (Dutta, Datta, VanderMeer, Thomas, & Ramamritham, 2007; Hjort, Lantz, Ericsson, & Gattorna, 2013; Lakew, Klein, Hernandez-rodriguez, & Elmroth, 2015). Internet service requests are normally classified into different categories based on their importance and treated differently. Services differentiation is achieved by assigning different priorities to the requests, and processing them in the order of their priorities. It is not uncommon that ISPs sometimes degrade the quality of delivered services in order to admit a larger number of requests (Chandra, Ellis, & Vahdat, 2000), provided the contracted SLAs are met.

Web mining has been researched extensively (Borges & Levene, 1999; Kim, Yeo, Koh, & Lipka, 2016; Siddiqui & Aljahdali, 2013; Totok & Karamcheti, 2010; Vanderveld & Pandey, 2016; Vijayalakshmi, Mohan, & Raja, 2010; Zhao, Sundaresan, Shen, & Yu, 2013). One of the web mining directions is web usage mining, which focuses on techniques for identification of user navigation behavior within a website. Web usage mining has many applications such as website design (Srivastava, Cooley, Deshpande, & Tan, 2000), user classification (Totok & Karamcheti, 2010), production recommendation (Jiang, Zhu, Zhang, & Yuan, 2015), purchase prediction (Liu et al., 2016; Yeo, Kim, Koh, Hwang, & Lipka, 2017), market analysis and advertisement (Zhao et al., 2013), and so forth. This paper focuses on web usage mining for revenue maximization. In an e-commerce application, final purchase is considered the most important request as it directly contributes to company revenue. However, other requests still need to be considered as they can also lead to final purchase; any frustration due to long response time can potentially result in the user leaving the site and loss of revenue, therefore good web mining can help understand user navigation behavior, improve workload management, and ultimately increase company revenue.

The idea in this paper is to use a data mining technique namely *association rule* to identify user navigation patterns within an e-commerce application, and to differentiate the relative importance of the requests by predicting how likely they will lead to final purchase. The idea of our approach takes the following into consideration: (a) only final purchase requests result in revenue contribution; (b) any other request can potentially lead to final purchase, depending on the likelihood of user navigation sequence starting from current request and leading to final purchase; (c) service differentiation and priority assignment are based first on aggregated *confidence* and then on average *support* of the composite association rules.

The main contribution of the work in this paper includes: (a) identification of composite association rules in typical e-commerce applications that capture user navigation patterns; (b) development of algorithms based on the identified composite association rules for request admission, service differentiation and priority scheduling in server overloading situations.

The paper is organized as follows. First, the related work is reviewed; then the navigation model of a typical e-commerce application is introduced; after that, the derivation of the association rules are described in details; next, it explains how the association rules are used in development of the algorithms for AC, service differentiation and priority scheduling; performance evaluation is then conducted, results are discussed; finally, some discussions are covered and the paper is concluded.

2 | RELATED WORK

Both internet server resource management and web data mining have attracted extensive research in the literature. Internet resource management can be achieved through resource provisioning (Urgaonkar et al., 2005; Urgaonkar, Kozat, Igarashi, & Neely, 2010; Villela et al., 2007; Xue et al., 2008), AC (Ashraf et al., 2012; Elnikety et al., 2004; Khojasteh et al., 2015; Konstanteli et al., 2014; Tomas & Tordsson, 2013; Wu et al., 2012), service differentiation (Chandra et al., 2000; Dutta et al., 2007; Hjort et al., 2013; Lakew et al., 2015), and request scheduling (Zhu, Yang, Chen, Wang, & Yin, 2014). It is common practice to combine more than one of the techniques to achieve better results. Khojasteh et al. (2015), Urgaonkar et al. (2005) and Villela et al. (2007) uses queuing network model for resource allocation for internet applications in the clouds. It is common practice to employ both proactive and reactive approaches for long and short term resource provisioning (Urgaonkar et al., 2005). In the early work, Xue et al. developed dynamic server switching algorithm to allocate server resources at real-time based on the workload demands within a shared hosting environment (Xue et al., 2008). The algorithm works well only when the hosted applications have contrasting workload demands.

Urgaonkar et al. (2010) applied Lyapunov Optimization for AC, routing, and resource allocation. Ashraf et al. (2012) uses measured and predicted resource utilizations of a server to make AC decisions based on user sessions. Apart from "accept" or "reject" decisions, it also considers "deferment" mechanism to reduce total sessions to be rejected. The work in (Khojasteh et al., 2015) uses a probabilistic optimization model for AC in elastic clouds hosting horizontally scaling services; it takes into account eco-efficiency and cost, as well as affinity and anti-affinity rules possibly in place for the components that comprise the services. In (Wu et al., 2012), the authors considered users, Software-as-a-Service (SaaS) providers, Infrastructure-as-a-Service (IaaS) providers and various SLAs between them, and made AC and scheduling decisions from economic perspective. In (Liu, Wang, Sun, Zou, & Yang, 2013), the authors proposed an approach that uses genetic algorithm (GA) to find optimal solution to address the issue related to benefits and costs in cloud computing, and the results showed effectiveness of the approach in terms of virtual resource utilization, rate of return on investment and operation profit. The authors in (Tomas & Tordsson, 2013) applied fuzzy risk assessment for AC in complex cloud systems.

The authors in (Hjort et al., 2013) use customer segmentation based on their buying and returning behaviors to support differentiated service delivery in fashion e-commerce. Performance-based service differentiation has been discussed in (Lakew et al., 2015)—in server overloading situations, it dynamically decides which services and to what extension should be degraded, while maintaining applications' performance targets.

Web usage mining has been used for a variety of purposes, including customer behavior studies (Hjort et al., 2013; Vijayalakshmi et al., 2010), request prioritization and service differentiation (Totok & Karamcheti, 2010), recommendation (Jiang et al., 2015), prediction (Liu et al., 2016; Yeo et al., 2017), and e-commerce market analysis (Zhao et al., 2013). Among all, the work in (Totok & Karamcheti, 2010) is closest to the work in this paper. It uses a reward-driven approach for request prioritization for e-commerce web sites. It classifies users from their previous navigation patterns, and prioritize the subsequent requests in server overloading situations. The proposed work in this paper uses a data mining technique to identify common navigation patterns from existing customers, then applies developed algorithms to requests from all (existing or new) users, who can potentially contribute to company revenue. The service differentiation is not applied at user level but at request level so that critical requests from less favorable users are not easily rejected. In (Borges & Levene, 1999), the authors model user navigation patterns with consideration of their previous navigation activities using the Ngram concept for deciding the transition probabilities. This paper uses first order Markov Chain for transition probabilities without consideration of previous navigation activities; this is mainly to avoid performance overheads in back tracking the activities. It is important that in server overloading situations, any extra activities should be minimized.

3 | WEB NAVIGATION MODELING

In this paper, we model typical e-commerce workloads using a weighted directed graph G = (N, E), where N is a set of nodes that represent user requests, and E is a set of edges that represent navigations between requests (Figure 1). Six typical e-commerce requests ("browse," "account," "logon," "logoff," "add to cart," and "purchase") are used in the navigation model, and six letters ("B," "A," "O," "F," "C," and "P") are used to denote the requests, respectively. The initial weights of the navigation model are assigned based on realistic values and are regularly updated using statistics in server logs thereafter.

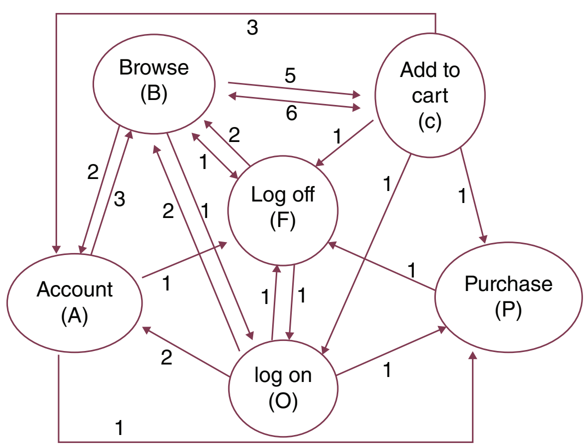

FIGURE 1 User navigation model for a typical e-commerce application. It is modeled using a weighted direct graph with nodes representing the typical web items and directed edges representing user navigation from one web item to another; the weights of edges are regularly updated based on statistics in server logs

Information that can be extracted from server logs includes user identification, the sequence of requested pages/links, the date and time of access, and so forth.

4 | IDENTIFICATION OF ASSOCIATION RULES

An association rule $r = (A \rightarrow B)$ in data mining describes how some items X related to other items Y in transactions, and is usually denoted as $X \Rightarrow Y$. A single association rule has *confidence*, that represents the likelihood of the rule and is denoted as C_r, and *support* that represents number of traversals to the link and is denoted as S_r.

A user session in an e-commerce application comprises of a sequence of requests, navigating the system. Mining user navigation patterns in such a system can be viewed as a generation of association rules. Let A_1, A_2, ...A_n denote a sequence of navigation activities, a composite association rule, denoted as $r = [A_1 \rightarrow A_2 \rightarrow A_3 \ldots A_{n-1} \rightarrow A_n]$, describes the *confidence* that the user will follow the order of the activities. The *confidence* of a composite association rule C_r is defined as the product of the *confidences* of all the corresponding single rules, that is, $C_r = \Pi_{i=1}^{n-1} C_i(A_i \rightarrow A_{i+1})$. The *support* of rule r, denoted as S_r represents the average number of times the links of r were traversed over the average number of times all of the links in the graph were traversed as described below:

$$S_r = \frac{\sum_{i=1}^{n-1} |(A_i, A_{i+1})| / (n-1)}{\sum_{i|(x_i, x_{i+1}) \in E} (x_i, x_{i+1}) / |E|}$$

As mentioned earlier, a user session in an e-commerce system consists of a sequence of requests that interact with the system. This paper mines user navigation patterns by using composite association rules to identify the subset of user navigation patterns that have higher probability of leading to final purchase. Calculation of the *support* and *confidence* values is conducted for each of the rules. The *Modified Depth-First Search* algorithm (Borges & Levene, 1998) is used in this paper to generate the composite association rules from the navigation model in Figure 1.

It is assumed that a user session, which contributes to company revenue, consists of at least *browse (B)*, *add to Cart (C)*, *logon (O),* and *purchase (P).* Also, a properly designed e-commerce application should not allow invalid requests (e.g., purchase without items in shopping cart), this helps session invalidation in a user navigation model. From the rules in Table 1, it can be seen that there are some direct links between some nodes and the final purchase (O) node, therefore it is assumed that there were previous navigation steps. It is also observed that when two rules have same requests in the same direction, the longer rule will have smaller *confidence* value. However, the *support* values can increase (e.g., $B \rightarrow O \rightarrow P$ and $C \rightarrow B \rightarrow O \rightarrow P$) or decrease (e.g., $B \rightarrow C \rightarrow P$ and $A \rightarrow F \rightarrow B \rightarrow C \rightarrow P$) as the length of a rule increases, depending on the weights of the involving links.

Once the association rules are identified, they can be used for algorithm development for AC and service differentiation. As we know that only final purchase requests result in revenue contribution, any other requests can potentially lead to final purchase, depending on the likelihood of the navigation sequence that starts from current request and ends at final purchase. Admission and service differentiation in this paper are only based on current and its next requests, which can be easily obtained from HTTP request headers. Given any request and its next request, the probability of the request reaching the final purchase stage can be obtained from Table 2, AC and differentiation decision can be made based on that. The aggregated *confidence* and *support* values of the composite association rules are recomputed based on updated weight values of the navigation model. Since they (i.e., the initial ones and those after each update of navigation weights in the model) are computed offline (perhaps on another computer), and they are in use until the new values are available, the proposed approach incurs minimum performance overheads.

5 | ADMISSION CONTROL

When a server is overloaded, quality of service (QoS) will inevitably become poor. In the worst case, the service will become unavailable. In such a situation, AC is necessary for busy ISPs to maintain the QoS. It works by rejecting some less important requests to maintain the overall response time to an acceptable level. ISPs may give their customers compensation for the rejected requests, depending on the contracted SLAs. From Table 1, we can calculate the aggregated *confidence* and average *support* values for all direct edges that can potential lead to final purchase in the navigation model. In this paper, an AC algorithm is developed based on the derived association rules. In Algorithm 1, an initial aggregated *confidence* threshold C_t is assigned based on realistic value; for every incoming request $r_{i \rightarrow j}$ that navigates from link i to j, the aggregated *confidence* C_r is obtained from Table 2; if $C_r < C_t$, then r will be rejected; if the mean response time M_{mrt}

TABLE 1 Derived composite association rules from user navigation model

Rules	Confidence	Support
$A \to P$	0.2	0.53
$A \to B \to C \to P$	0.028	1.58
$A \to B \to C \to O \to P$	0.0046	1.32
$A \to B \to C \to F \to O \to P$	0.0015	1.16
…	…	…
$O \to F \to B \to C \to P$	0.005	0.79
$O \to F \to B \to C \to A \to P$	0.003	1.27

of the system is longer than the agreed response time M_{sla}, then C_t will be increased; this process will be repeated until C_t reaches an appropriate level.

Algorithm 1: Association rules based admission control

Input: incoming requests
Output: admission control decision
Initialization: assign initial threshold *confidence* C_t
for each incoming request $r_{i \to j}$ from i to j **do**
  Obtain aggregated *confidence* C_r from Table 2
  if $C_r < C_t$ **then**
    reject r
  end if
  if $T_{mrt} > T_{sla}$ **then**
    increase C_t
  else if $T_{mrt} < T_{sla}$ **then**
    decrease C_t
  end if
end for

Algorithm 2: Admission control and priority scheduling algorithm

Input: incoming requests (r, s, ...), *SLA*
Output: priority queue
for each incoming request r and s **do**
  Get aggregated *confidence* C_r and average *support* S_r;
  Assignment priority P_r to request r based on SLA_r and/or C_r/S_r;
  if $P_r < P_s$ **then**
    place r after s in the priority queue
  end if
end for
while $T_{mrt} > T_{sla}$ **do**
  reject requests at the end of the priority queue
end while

TABLE 2 The list of calculated aggregated *confidence* and average *support* values from any single node to final purchase represented as 'O' in Figure 1

Link	Aggregated confidence	Average support
$A \to B$	0.0341	1.35
$A \to F$	0.0205	0.86
$A \to P$	0.2	0.53
$B \to A$	0.0465	0.73
…	…	…
$O \to B$	0.0274	1.25
$O \to F$	0.013	1.08
$O \to P$	0.17	0.53

6 | SERVICE DIFFERENTIATION

The AC algorithm described in last section ensures the number of requests in the system remains at an appropriate level. However, the algorithm does not differentiate admitted requests, that is, all of them are processed in a FIFO order. Due to the nature of e-commerce applications and various SLAs between companies and ISPs, some requests are considered more important than others (e.g., a browsing request is less important than a purchase request, the requests from customers who pay higher fees are considered more important than others); therefore, some requests should be given higher priority than others. In this paper, priority assignment is based on aggregated *confidence* and average *support* values as listed in Table 2. When two requests have identical aggregate *confidence* values, the one with higher average *support* value will be assigned higher priority. All incoming requests will be placed in a priority queue based on their priorities and requests with highest priorities will be processed earlier. To maintain the overall response time at an appropriate level, AC is also applied by rejecting requests at the end of priority queue (see Algorithm 2). The combined approach ensures that the most important requests will get the best service (i.e., shortest mean response time) while maintaining the overall response time at a satisfactory level so that all SLAs can be achieved.

7 | EVALUATION

In order to evaluate the performance of the proposed approach, several experiments have been conducted for the following approaches: (a) standard approach (No_AC_No_PS) without admission control and priority scheduling; (b) priority scheduling based on SLAs, no Admission Control (No_AC_PS_SLA); (c) combination of admission control and priority scheduling based on identified association rules (AC_PS_AR); (d) combination of admission control and priority scheduling based on SLAs (AC_PS_SLA); (e) combination of admission control and priority scheduling based on both association rules and SLAs (AC_PS_AR_SLA). The simulation generates 1,000 random web users, each of which will have a sequence of web requests. The navigation sequences are based on the transition probabilities calculated from the navigation model in this paper. The performance comparison is based on the total revenue units generated from the approaches. The association rules in Table 2 are used to set request priorities—based on current request and the next request, the higher the aggregation confidence, the higher the priority of the next request; when aggregated confidence is the same, aggregated support will be used for determining the priority of the next request. In this paper, we also simulate priority scheduling for those *premier* users, whose requests have highest priority as different fees have been charged; the simulations in this paper generates 10% of the premier users. In our simulation, when a request reaches final purchase, it results in one unit of revenue; when a customer leaves the cite without reaching the final purchase stage, or a request is rejected, no revenue will be generated. In the simulation, the probabilities in the navigation model are fixed, however, they can be slightly different overtime, therefore the model should be updated periodically based on statistics in the web logs. As processing of the log files and computation of navigation probabilities can be done offline, it incurs minimum performance overheads, therefore, the interval in between model updates can be set to a small value for better accuracy of the model. Also, the actual navigation patterns in real world e-commerce applications are likely to be different to the navigation model in this paper (e.g., different web items and links in between), the model needs to be adjusted accordingly.

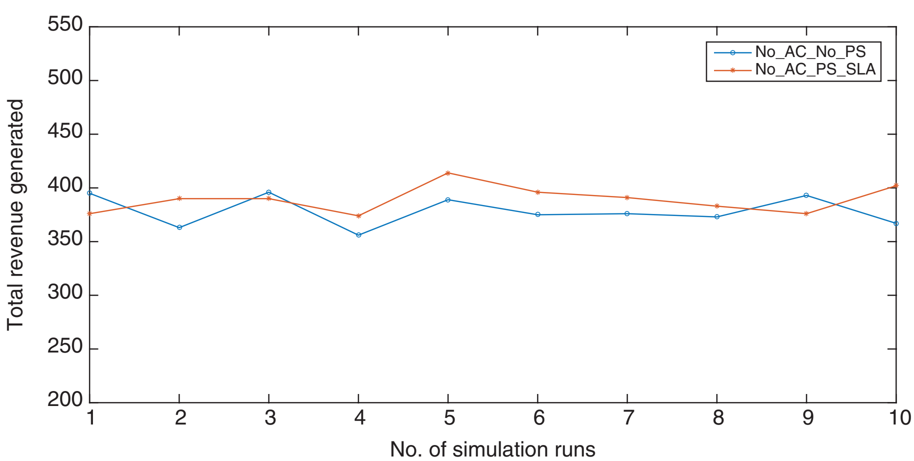

FIGURE 2 Comparison between two approaches in terms of revenue generation: (1) No priority scheduling, no Admission Control; (2) Priority Scheduling based on SLA, No Admission Control

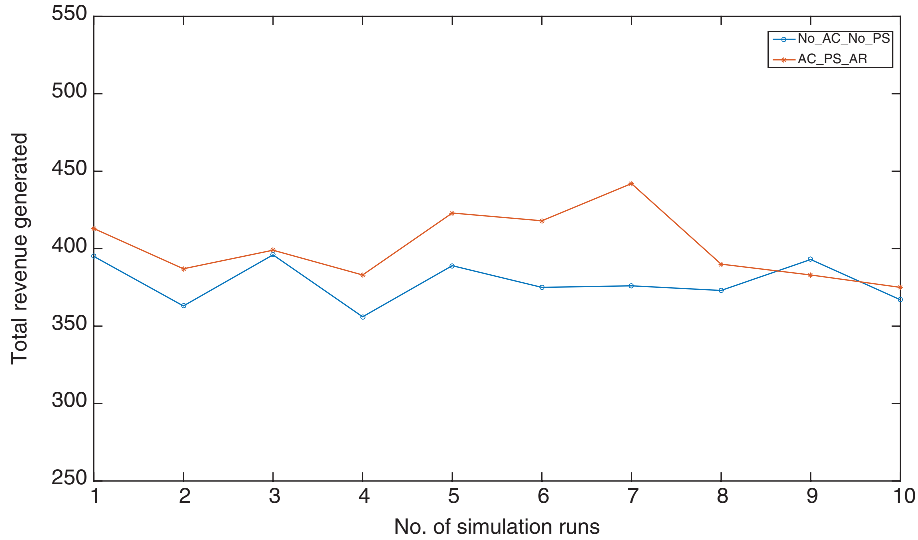

FIGURE 3 Comparison between two approaches in terms of revenue generation: (1) Neither priority scheduling nor Admission Control is applied; (2) Combination of Admission Control and Priority Scheduling based on Association Rules is applied

As can be seen from Figure 2, when there is no AC, but priority scheduling based on SLAs is applied, the improvement in revenue is approximately 3%; however, if AC and priority scheduling based on association rules are applied (Figure 3), the improvement is increased to about 6%; when AC and priority scheduling based on SLAs are used (Figure 4), the generated revenue units are about 15% better; finally, when combination of AC and priority scheduling based on both SLAs and association rules in Table 2 (Figure 5), there is a significant improvement in revenue units (about 66%).

The simulation results show that the proposed approach works effectively. The performance evaluation provides some numeric evidence for both cloud resource providers and e-commerce companies to review the SLAs in terms of QoS and pricing models. In a fixed fee model, AC will be applied to maintain QoS and ensure appropriate resource allocation; the pay-as-you-go model makes use of elasticity of cloud computing, resources will be allocated according to workload levels of individual e-commerce applications, but no preferential processing would take place; for those e-commerce companies willing to pay higher fees, service differentiation will be applied to ensure best QoS.

8 | DISCUSSION AND CONCLUSIONS

The workload demands in e-commerce applications are very unpredictable. In addition to usual fluctuation, from time to time there are big surges, which impose big challenges to internet resource providers. Static resource provision techniques do not work well, as they can result in either waste of resources or insufficient resources being allocated. Dynamic resource provisioning can solve the problem to certain extent; however, both starting up new servers and switching server resources between internet applications take time, therefore prior knowledge about workload patterns for long-term proactive planning and fast response to short-term workload changes are critical in dynamic resource provisioning.

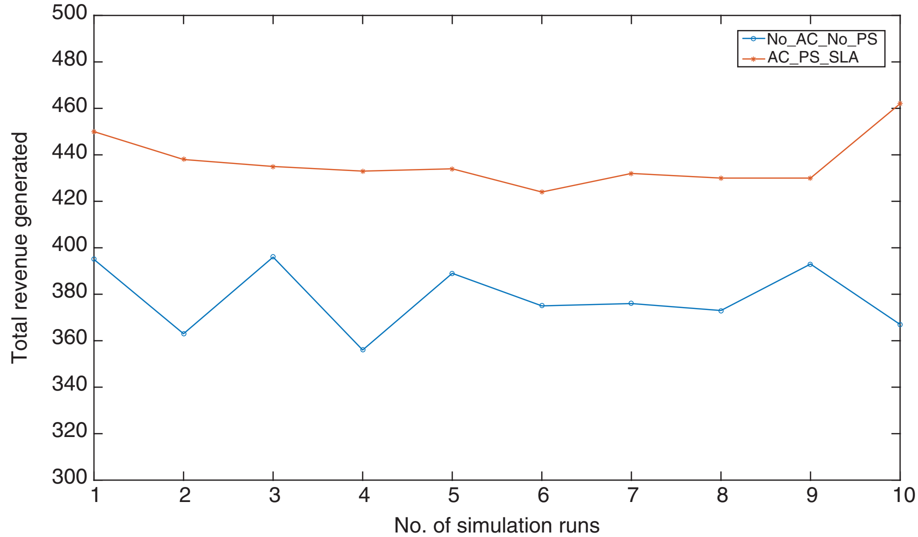

FIGURE 4 Comparison between two approaches in terms of revenue generation: (1) No priority scheduling, no Admission Control; (2) Combination of Admission Control and Priority Scheduling based on SLAs

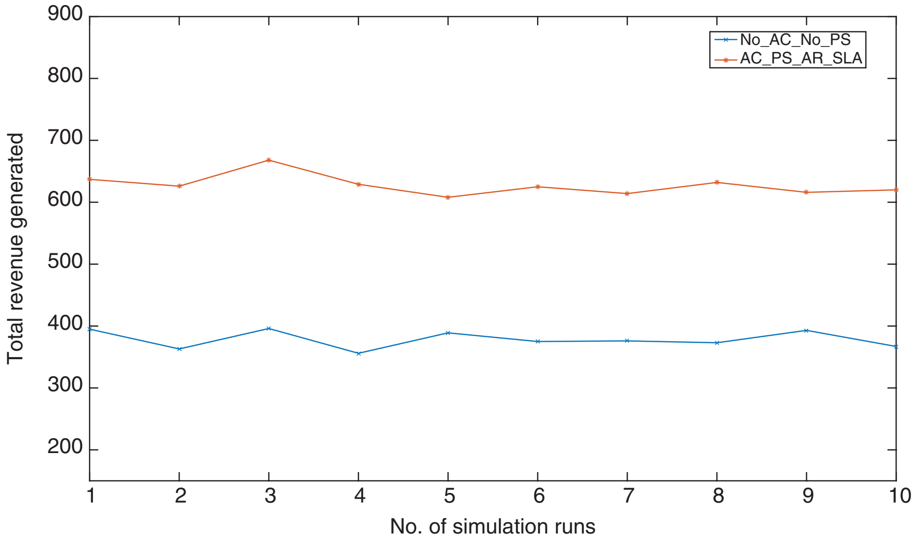

FIGURE 5 Comparison between two approaches in terms of revenue generation: (1) No priority scheduling, no Admission Control; (2) Combination of Admission Control and Priority Scheduling based on both SLA and Association Rules

Elastic cloud computing offers horizontal scalability to e-commerce applications. Underpinning cloud computing are the virtualization techniques, which enable seamless creation and migration of virtual instances within the Cloud. Virtual instances within clouds can grow and shrink dynamically in response to workload changes.

In light of elastic cloud computing, one might argue that resource provisioning and AC are no longer needed as cloud resources can grow and shrink on demand. This is usually not the case. First of all, cloud infrastructure providers need to make the best of their resources to maximize the profits. They offer services to large number of customers with various workload patterns, resource provisioning and management become an even more challenging task. Moreover, there are a variety of SLAs between cloud resource providers and their customers. While some customers might pay for extra resources due to workload surges, whereas others just pay a fixed amount of fees, therefore AC is needed to ensure SLAs are met and appropriate amount of resources is allocated.

Web stream clicks in e-commerce applications are usually stored in log files on servers. However, the log files cannot capture all the information as some web responses are generated directed from caches in proxy servers. In addition, some service providers use cookies to store and track user information on their browsers. The quality of web usage mining depends on collection of user information and data mining techniques. Association rule is a simple but powerful data mining technique, and is commonly applied in e-commerce applications for production recommendation. In this paper, the author models customer behavior in a typical e-commerce application using a weighted direct graph. The initial weights of the navigation model are assigned based on realistic values and regularly adjusted using information in server logs. A modified depth first search (DFS) algorithm is then applied to identify all composite association rules that represent user navigation within the system. Two algorithms are then developed from the identified association rules, and applied for AC, service differentiation and priority scheduling. The performance evaluation results show that the combination of AC and priority scheduling using our approach is effective in terms of workload management and revenue maximization. Our approach is easy to implement and can be adopted by cloud-based web hosting companies.

CONFLICT OF INTEREST

The authors have declared no conflicts of interest for this article.

FURTHER READING

Amazon. (2017). *Amazon web services auto scaling*. Retrieved from http://aws.amazon.com/autoscaling

REFERENCES

Ashraf, A., Byholm, B., & Porres, I. (2012). A session-based adaptive admission control approach for virtualized application servers. In *5th IEEE International Conference on Utility and Cloud Computing (UCC)* (pp. 65–72).

Borges, J., & Levene, M. (1998). Mining association rules in hypertext databases. In *Proceedings of Fourth International Conference on Knowledge Discovery and Data Mining* (pp. 149–153).

Borges, J., & Levene, M. (1999). Data mining of user navigation patterns. In *Proceedings of the WEBKDD'99 Workshop on Web Usage Analysis and User Profiling* (pp. 31–36).

Chandra, S., Ellis, C. S., & Vahdat, A. (2000). Differentiated multimedia web services using quality aware transcoding. In *Proceedings of IEEE INFOCOM 2000*.

Dutta, K., Datta, A., VanderMeer, D., Thomas, H., & Ramamritham (2007). ReDAL: An efficient and practical request distribution technique for application server clusters. *IEEE Transactions on Parallel and Distributed Systems*, *18*(11), 1516–1527.

Elnikety, S., Nahum, E., Tracey, J., & Zwaenepoel, W. (2004). A method for transparent admission control and request scheduling in e-commerce web sites. In *International WWW Conference*, New York, USA.

Hjort, K., Lantz, B., Ericsson, D., & Gattorna, J. (2013). Customer segmentation based on buying and returning behaviour. *International Journal of Physical Distribution & Logistics Management*, *43*(10), 852–865.

Jiang, P., Zhu, Y., Zhang, Y., & Yuan, Q. (2015). Life-stage prediction for product recommendation in e-commerce. *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining.*

Khojasteh, H., Misic, J., & Misic, V. B. (2015). Task admission control for cloud server pools. In *IEEE Wireless Communications and Networking Conference (WCNC).*

Kim, S., Yeo, J., Koh, E., & Lipka, N. (2016). Purchase influence mining: Identifying top-k items attracting purchase of target item. *WWW16 Companion, Canada.*

Konstanteli, K., Cucinotta, T., Psychas, K., & Varvarigou, T. (2014). Elastic admission control for Federal Cloud Services. *IEEE Transactions on Cloud Computing*, *2*(3), 348–361.

Lakew, E. B., Klein, C., Hernandez-rodriguez, F., & Elmroth, E. (2015). Performance-based service differentiation in clouds. *In 15th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid)* (pp. 505–514).

Liu, G., Nguyen, T. T., Zhao, G., Yang, J., Wu, M., Zhao, P., & Chen, W. (2016). Repeat buyer prediction for e-commerce. In *KDD'16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Liu, Z., Wang, S., Sun, Q., Zou, H., & Yang, F. (2013). Cost-aware cloud service request scheduling for SaaS providers. *The Computer Journal*, *57*(2), 291–301.

Siddiqui, A. T., & Aljahdali, S. (2013). Web mining techniques in E-commerce applications. *International Journal of Computer Applications.*, *69*(8), 39–43.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. N. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, *1*(2), 12.

Tomas, L., & Tordsson, J. (2013). Cloudy with a chance of load spikes: Admission control with fuzzy risk assessments. In *Proceedings of 6th IEEE/ACM International Conference on Utility and Cloud Computing* (pp. 155–162).

Totok, A., & Karamcheti, V. (2010). RDRP: Reward-driven request prioritisation for e-commerce web sites. *Electronic Commerce Research and Applications.*

Urgaonkar, B., Shenoy, P., Chandra, A., & Goyal, P. (2005). Dynamic provisioning of multitier Internet applications. In *Second International Conference on Autonomic Computing (ICAC'05).*

Urgaonkar, R., Kozat, U. C., Igarashi, K., & Neely, M. J. (2010). Dynamic resource allocation and power management in virtualized data centers. In *Proceedings of IEEE/IFIP NOMS.*

Vanderveld, A., & Pandey, A. (2016). An engagement-based customer lifetime value system for e-commerce. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining.*

Vijayalakshmi, S., Mohan, V., & Raja, S. S. (2010). Mining of users' access behaviour for frequent sequential pattern from web logs. *International Journal of Database Management Systems (IJDMS)*, *2*(3), 31–45.

Villela, D., Pradhan, P., & Rubenstein, D. (2007). Provisioning servers in the application tier for e-commerce systems. *ACM Transaction on Internet Technology*, *7*(1), 7.

Wu, L., Garg, S. K., & Buyya, R. (2012). SLA-based admission control for a Software-as-a-Service provider in cloud computing environments. *Journal of Computer and System Sciences*, *78*(5), 1280–1299.

Xue, J., Chester, A. P., He, L., & Jarvis, S. A. (2008). Dynamic resource allocation in enterprise systems. In *14th International conference on Parallel and Distributed Systems (ICPADS'08).*

Yeo, J., Kim, S., Koh, E., Hwang, S., & Lipka, N. (2017). Predicting online purchase conversation for retargeting. In *ACM WSDM'17.*

Zhao, Y., Sundaresan, N., Shen, Z., & Yu, P. S. (2013). Anatomy of a web-scale resale market: A data mining approach. In *ACM WWW'13* (pp. 1533–1544).

Zhu, X., Yang, L. T., Chen, H., Wang, J., & Yin, S. (2014). Real-time tasks oriented energy-aware scheduling in virtualised clouds. *IEEE Transactions on Cloud Computing*, *2*(2), 168–180.

How to cite this article: Xue J, Jarvis S. Mining association rules for admission control and service differentiation in e-commerce applications. *WIREs Data Mining Knowl Discov*. 2018;8:e1241. https://doi.org/10.1002/widm.1241