Applied Mathematics & Information Sciences
An International Journal

A New Method for User Dynamic Clustering Based On HSMM in Model of SaaS

ChunHua Ju[1,2] *and Chonghuan Xu*[3,*]

[1]College of Computer Science & Information Engineering, Zhejiang Gongshang University, HangZhou 310018 P. R. China
[2]Center for Studies of Modern Business, Zhejiang Gongshang University, HangZhou 310018 P. R. China
[3]Business Administration college, Zhejiang Gongshang University, HangZhou 310018 P. R. China

Received: 22 Nov. 2012, Revised: 15 Jan. 2013, Accepted: 27 Jan. 2013
Published online: 1 May 2013

Abstract: This paper deeply studies the phenomenon of hard to satisfy the user's personalized services and only a few researches on users themselves in the model of Software as a Service (SaaS), then proposes a users' behavior feature extraction model based on Hidden Semi-Markov Models (HSMM) to solve the problem of getting users hidden information on SaaS platform first. The model uses the probability distribution of state duration time to control user's browsing behaviors, combines hidden states which describe features with time relativity, and applies improved Viterbi algorithm to get user features sequence. Then cluster users by dynamic K-means algorithm, which doesn't need to give K cluster centers in the process of clustering but adjusts center value automatically through the comparison of clustering quality in every iterative process, finally gets optimal clustering results. Detailed simulation analysis demonstrates that the presented algorithm is of high efficiency of space and time and is more stable.

Keywords: Software as a Service, User Characteristics, Hidden semi-Markov Model, Dynamic Clustering

1. Introduction

Cloud computing and cloud services have recently become hot issues in improving organizations' information technology (IT) competitiveness and performance. Cloud computing involves making computing, data storage, and software services available via the Internet [1]. Goscinski and Brock [2] indicated that computing resources hosted within the cloud can perform in many roles such as database services, virtual servers, service workflows or configurations of distributed computing systems. More importantly, cloud services based on cloud computing can free an organization from the burden of having to develop and maintain large-scale IT systems; therefore, the organization can focus on its core business processes and implement the supporting applications to deliver the competitive advantages.

Generally, cloud services can be divided into three subcategories: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Among them, SaaS [3] is regarded as a potential segment and the utilization of SaaS solutions can lead to many benefits for enterprise users with profound consequences in improving IT performance. SaaS pushes the functions of common software onto the infrastructure layer and achieves various business functions on it, third-party operators are responsible for operation and maintenance, so the security and reliability of data depend completely on the credibility of service operators. SaaS provides a pathway with fast, convenient implementation and advanced informationization technology to users, but usually, it only offers standardization and unification system services instead of satisfying users' personalized services.

At present, the studies on personalized services in SaaS model are still in an exploratory stage and there are few literatures in this field at home and abroad[4]. Wu et al.[5] presumed that an organization will augment the trust of adopting SaaS solutions when perceived risks decrease and/or perceived benefits increase. To gain insights into this issue, a solution framework using a modified Decision Making Trial and Evaluation Laboratory (DEMATEL) approach is proposed. Wu et al. [6] also developed an explorative model that examines important factors affecting SaaS adoption, in order to facilitate understanding with regard to adoption of SaaS solutions. An explorative model using partial least

* Corresponding author e-mail: talentxch@gmail.com

squares (PLS) path modeling is proposed and a number of hypotheses are tested, which integrates TAM related theories with additional imperative constructs such as marketing effort, security and trust. On the basis of study on SaaS business process customization and authentication mechanisms of TLA, Shi et al.[7] proposed a behavior model and authentication framework which supports tenant business process customization and designs services recommendation algorithm to satisfy different tenants' personalized business services. Zhang et al.[8] proposed a policy-driven customization mechanism, if the tenant's customization demand is the same with customization policy, the provider will upgrade the services. Wu et al.[9] proposed an analytical framework containing two approaches-Technology Acceptance Model (TAM) and Rough Set Theory (RST). An empirical study on the IT/MIS enterprises in Taiwan is carried out. The results have revealed a considerable amount of meaningful information, which not only facilitates the SaaS vendors to grasp users' needs and concerns about SaaS adoption, but also helps the managers to introduce effective marketing strategies and actions to promote the growth of SaaS market. Jaeger et al. [**?**, 10]sed mode decision algorithm to determine every process node's best service. Kim et al.[11] proposed a novel method for integrating existing softwares in the SaaS environment. This method can be applied to all the commercial softwares, and they illustrated business software integration using the proposed method. The integrated software is service-oriented through Internet access, where the customers only pay for the service that they want to use.

This paper focuses on users themselves and proposes an integration model based on Hidden Semi-Markov Models (HSMM) for user's behavior features extraction of SaaS platform. HSMM uses the probability distribution of state duration time to control user's browsing behaviors, combines hidden states which describe features and time relativity tightly. Also it can generate multiple observation sequences and according to this characteristic, it divides text messages into multiple text block subdomains to make every subdomain's feature be mutual correspondence to an observation sequence, then applies improved Viterbi algorithm to get user features sequence. Later, cluster users based on user features sequence. Here, we propose a dynamic K-means clustering algorithm to cluster. By the simulated test, this model and algorithm were showed great effectiveness and practicability. A review of background is given in Section 2. In Sections 3, the way our method of user's behavior features extraction is described. Section 4 describes the dynamic clustering algorithm based on HSMM. Section 5 provides experimental results of presented algorithm on a real dataset. Finally, we conclude in Section 6.

2. Background

2.1. Hidden Markov Model

Hidden Semi-Markov Models (HSMM)[12–15] is a semi-continuous HMM between continuous and discrete Hidden Markov Model, it allows the basic process to be a semi-Markov chain and has a variable cycle or residence time for each state. A HSMM with N states can be briefly denoted as $\lambda = (N, M, p, A, B, p_i(d))$, its model is shown in Figure 1.

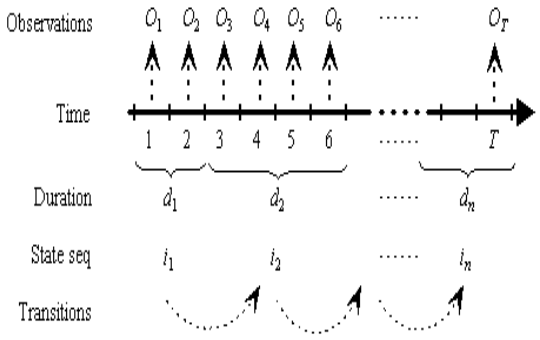

Figure 1 HSMM model

2.2. Characteristic of coincidence

The different degree of coincidence between two users' data streams exists in multiple data streams environment of SaaS platform. The degree of data stream coincidence reflects the similarity of variation tendency of data stream. The essence of multiple data streams clustering is to gather similar objects through analyzing the similarity of data streams.

Defintion 1. For the data streams $X = \{x_1, x_2, ..., x_n\}$,$Y = \{y_1, y_2, ..., y_n\}$.Given the formula:

$$\rho_{xy} = \sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y}) / \sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2 (y_i - \bar{y})^2} \qquad (1)$$

and $\bar{x} = (x_1 + x_2 + ... + x_n)/n$,$\bar{y} = (y_1 + y_2 + ... + y_n)/n$, in which ρ_{xy} reflects the similarity between X and Y, also $|\rho_{xy}| \leq 1$, bigger the ρ_{xy} is, higher the relevance is.

3. The method of user's behavior features extraction

Detailed steps: at first, preprocess the marked samples and use BW algorithm to train HSMM, initialize and

build the most appropriate HSMM, then introduce the preprocessed features to the model and use the improved Viterbi algorithm to decode, thus output required marked sequence which exactly are user's behavior features needed.

3.1. Model establishment and features extraction

(1) Preprocess the feature information of user behaviors orderly, select keywords, titles, page retention time, marked behaviors, operation behaviors and link behaviors as hidden state sequences, and subdivide the later three ones to the next level of hidden state, for example, marked behavior contains the increase of bookmarks and the save of pages these two hidden state sequences.

(2) Preprocess marked training samples, collect server and client's data, form texts after preliminary treatment and scan the texts, then transform marked text sequence into marked text block sequence on the basis of typesetting and separator information such as newline, colon, two lines spaces and so on.

(3) Calculate HSMM's parameters.

(4) Apply established HSMM to extract user's behavior features. Take the processed observations $O = O_1, O_2, ..., O_t$ for the input of the model, then use improved Viterbi algorithm to calculate and find the marked state sequence of maximum probability, the observation text marked as target state label is user's features which have been extracted.

3.2. Improved Viterbi algorithm

Viterbi algorithm is used to find the optimal state sequence $O^* = q_1^*, q_2^*, q_t^*, ...,$ and maximize the probability $P(Q, O|\lambda)$. As in the process of iteration of Viterbi algorithm, underflow problem and time complexity will increase rapidly with the increase of constraint length caused by too much continual multiplication, for this shortcoming, we propose an improved method whose specific measures are as follows: use logarithm method to solve the problem of underflow and the rapid increase of time complexity in HSMM, modify the main formulas involved in the steps of Viterbi algorithm.

(1) Preprocessing: $\pi_i' = log(\pi_i), b_i'(o_t) = log(b_i(o_t)), a_{ij}' = log(a_{ij}), p_j' = log(p_j(d))$.

(2) Initialization: partial probability is $\delta_1(i)' = log(\delta_1(i))$, backward pointer is $\phi_1(i) = 0$.

(3) Recursive process: partial probability is

$$\delta_t'(i) = log[\delta_t(i)]$$

$$= max_{1 \le i \le N}\{log[\sum_{d-1}^{t} \delta_{t-d}(i) a_{ij} p_j(d) \prod_{s=t-d+1}^{t} b_j(O_s)]\},$$

backward pointer is $\phi_t(i) = argmax_{1 \le i \le N}\{\delta_{t-1}(j) a_{ji}\}$

4. Simulation model and algorithm

This section proposes a users dynamic clustering algorithm, dynamic K-means, which clusters the users on the basis of user features sequence mentioned above. The whole process of users dynamic clustering in SaaS is shown in Figure 2.

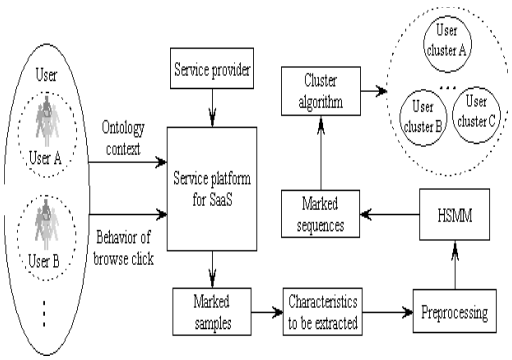

Figure 2 Flow diagram of users dynamic clustering in SaaS

4.1. The algorithm

In this paper, we set reciprocal of ρ_{xy} as measurement of distance and use objective function $G = \sum_{i=1}^{K} \sum_{x_j \in c_i} 1/\rho_{x_j c_i'}$ to dynamically evaluate the clustering quality of each iteration. Bigger the G is, better the cluster quality is, and set c_i' as the center of cluster c_i ,$\rho_{x_j c_i'}$ as the correlation coefficient of data stream x_j and corresponding cluster center.

Dynamic K-means algorithm is a method of optimized solution, distinguishes of K-means algorithm, it doesn't need to initialize K cluster centers, but gets optimal number of clusters K_{opt} through calculations instead. At first, we set $K = (\sqrt{n} - 1)/2$ (references the method of selection of K in literature [16], which satisfies with the condition of $K_{opt} \le K_{max}$ and $K_{max} \le \sqrt{n}$, and we consider the strategy of binary search algorithm further), calculate the clustering qualities of $K-1$, K and $K+1$, adjust the number of K, and iterate until K cluster centers don't change.

In detail: in the first division, according to the value of K, a data stream x_i is randomly selected as a center stream, then select other $K-1$ data streams which have the smallest correlation to it as the $K-1$ cluster centers. Then the remaining data streams are put into these K clusters based on the measure of the closest correlation, and calculate the clustering quality G_K.

For the initial clusters which have been clustered, two kinds of operation are taken: 1) a data stream is selected

randomly from the clustered clusters as a new added cluster center, then cluster and calculate G_{K+1} again; 2) cluster and calculate G_{K-1} again after removing a center. After that, compare G_{K-1}, G_K and G_{K+1}, get the biggest G, that is, the best clustering quality in this step. So we can choose the corresponding K as the number of cluster centers for next calculation, for example, if the clustering quality of $K-1$ clusters is the best, the number of clusters is $K-1$. Keep on until the clustering quality adjusts to the optimal.

Now we have selected the best number of clusters K_{opt}, then calculate the mean value of each cluster which have been clustered, and set these mean values as new centers, then divide clusters again, and adjust until there is no change according to the clustering steps of K-means. The Pseudo code of dynamic K-means is as follows:

The model demonstrates that the financial stress index trend could be simulated properly through the explanatory variables lagging four quarters (two other variables are lagging six quarters and eight quarters respectively). In other words, according to the current financial stress index and other values of explanatory variables, the financial stress index in the last four quarters could be forecast.

Input: initial center K, data streams D

Output: stable cluster proposal R_k

1 begin Initialize
2 randomly selects x_i and gets $x_1,\ldots,x_{k-1}$; // $x_1,\ldots,x_{k-1}$ are the streams which have the smallest correlation with x_i;
3 for $x_j \in D$ do; // x_j is the remaining data stream
4 calculate the correlation of x_j and each other center, classify it into the closest correlation cluster center, and calculate G_K;
5 add a center x_{random}, and calculate G_{K+1}; in another operation, remove a center x_{random}, and calculate G_{K-1};
6 argmax(G_{K-1},G_K,G_{K+1}) and get the corresponding K ;
7 go on until we get K_{opt};
8 n accordance with the K-means, adjust cluster centers until they don't change, output the result R_k;

5. Experimental result

The program is written in Matlab under the Matlab 7.9 running on Windows server 2008. The tests were performed on a Core(TM) i7 2.67GHz with 4 GMB Memory and 500GB Hard disk. This experiment is divided into two parts: 1) test the efficiency of time and space of dynamic K-means; 2) test the user clustering effect based on HSMM feature extraction in SaaS. The data is derived from users' browsing behaviors and basic information on SaaS platform.

5.1. The experiment of dynamic K-means

The number of users' browsing behavior data is 32428, and each one contains user's ID, browsing address and browsing time. In the process of user clustering, we need to complement user's personal information, the key words of browsing page and so on, then process browsing path. This paper compares dynamic K-means with K-means on the basis of execution time and memory consumption. The results are shown in Figure 3.

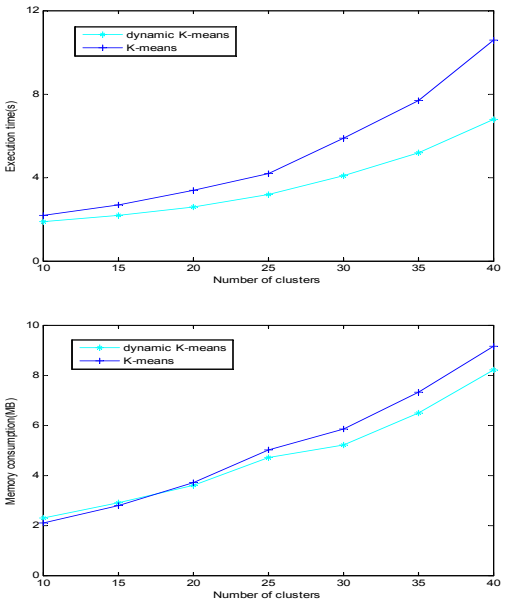

Figure 3 The comparison of time and space efficiency

We can know from Figure 3 that the execution time and memory consumption of dynamic K-means are better than K-means, especially when the number of data and clustering attributes is large.

5.2. The user clustering experiment in SaaS

The experimental data of SaaS platform include users' features information data and classification data. In order to test the users clustering effect based on HSMM feature extraction in SaaS, this paper designs three comparison experiments and uses external standards to test the anastomotic degree between clustering result and existing classification based on existing data structure, finally, analyses the results by classification accuracy, class precision and recall, in which Accuracy(AC), Precision(PE) and Recall(RE) are denoted as follows [17]:

$$AC = \sum_{i=1}^{k} a_i/n \quad (2)$$

$$PE = \sum_{i=1}^{k} (a_i/a_i + b_i)/k \quad (3)$$

$$RE = \sum_{i=1}^{k} (a_i/a_i + c_i)/k \quad (4)$$

n denotes the object number of datasets, a_i denotes the object number classified correctly in class i, b_i denotes the object number classified false in class i, c_i denotes the object number has not been classified in class i which should, k denotes the number of clusters.

Comparison experiment 1: compare K-means without users' behavior features extraction with K-means based on HSMM users' behavior features extraction. Table 1 shows the result of comparison of clustering effect.

Table 1 The comparison of clustering effect

Validation Measure	K-means	K-means based on HSMM
AC	0.622	0.731
PE	0.606	0.713
RE	0.613	0.729

Comparison experiment 2: compare dynamic K-means based on HMM users' behavior features extraction with dynamic K-means based on HSMM users' behavior features extraction. Table 2 shows the result of comparison of clustering effect.

Table 2 The comparison of clustering effect

Validation Measure	dynamic K-means based on HMM	dynamic K-means based on HSMM
AC	0.732	0.741
PE	0.716	0.720
RE	0.719	0.734

Comparison experiment 3: compare K-means based on HSMM users' behavior features extraction with dynamic K-means based on HSMM users' behavior features extraction. Table 3 shows the result of comparison of clustering effect.

Table 3 The comparison of clustering effect

Validation Measure	K-means based on HSMM	dynamic K-means based on HSMM
AC	0.731	0.741
PE	0.713	0.720
RE	0.729	0.734

We can know from the comparison experiments that the users clustering effect of dynamic K-means based on HSMM feature extraction proposed in this paper is better than traditional methods.

6. Conclusion

This paper deeply studies the users' personalized services in the model of SaaS and proposes a users' behavior features extraction model based on HSMM. For the shortcomings of Viterbi algorithm in HSMM, we present certain improvements and raise the efficiency of time and space. Besides, in order to conquer K-means's existing shortages, we propose a dynamic clustering algorithm called dynamic K-mean to cluster users, which doesn't need to give K cluster centers in the process of clustering, through the comparison of clustering quality in every iterative process, it adjusts center value automatically and then gets optimal clustering results. Experimental results on real and synthetic datasets show that our algorithm has high accuracy, efficiency and stability. For a future work, this paper suggests studies on how to decrease computational complexity and add knowledge rules to raise the precision of model.

Acknowledgement

This work was supported in part by NSFC of China under Grant No. 71071140 and 71071141, Research Fund for the Doctoral Program of Higher Education of China under Grant No. 20103326110001 as well as Zhejiang Provincial Natural Science Foundation of China under Grant No. Z1091224 and LQ12G01007,Key Innovation Team of Zhejiang Province (Grant No. 2010R50041).

The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] He.Heng, Li Ruixuan, Dong Xinhua, Applied Mathematics & Information Sciences **3**, 729-739 (2012).
[2] G.Andrzej, B.Michael, future generation computer systems. **26**, 947-970 (2010).
[3] W.Sun, K.Zhang, S.K.Chen. Lecture Notes in Computer Science. Springer-Verlag Publishers, Berlin 558-569 (2009).
[4] Sun Weifeng, Sun Mingyang, Liu Xidong, Applied Mathematics & Information Sciences **2S**, 69-78 (2011).
[5] W.W.Wu, L.W.Lan, Y.T.Lee, International Journal of Information Management **31**, 556-563 (2011).
[6] W.W.Wu, Expert systems with applications **38**, 15057-15064 (2011).
[7] Y.L.Shi, S.Luan, L.Qing, Q.Z.Li, J.L.Dong , F.F.Liu, Chinese Journal of Computer **33**, 2055-2067 (2010).
[8] K.Zhang, X.Zhang, W.Sun, H.Q.Ling , Y.Huang. Proceedings of the 9th IEEE International Conference on E-Commerce Technology (CEC) and the 4th IEEE International Conference on Enterprise Computing 123-130 (2007).
[9] W.W.Wu, Journal of Systems and Software 84, 435-441 (2011).
[10] M.C.Jaeger, G.Muhl , S.Golze. Proceedings of the 2005 IEEE International Conference on Web Services (ICWS). 807-805 (2005).
[11] W Kima, J.H Lee,C.L Hong, C.H Han, H.K Lee, B.S Jang. Computers & Mathematics with Applications **64**, 1252-1258 (2012).
[12] S.Z.Yu, Artificial Intelligence **174**, 215-243 (2010).
[13] J.Bulla, I.Bulla, O.Nenadic, Computational Statistics & Data Analysis **54**, 611-619 (2010).
[14] X.B.Tan, H.S.Xi, Applied Mathematics and Computation **205**, 562-567 (2008).
[15] X.D.Tan, J.Qiu, G.J.Liu, Q.H.Zeng, Q.Miao, Chinese Journal of Scientific Instrument **7**, 1341-1346 (2009).
[16] S.L.Yang, Y.S.Li, X.X.Hu, R.Y.Pan, Chinese Journal of Systems Engineering Theory & Practice **26**, 97-101 (2006).
[17] J.Y.Liang, L.Bai, F.Y.Cao, Chinese Journal of Computer Research and Development **47**, 1749-1755 (2010).

Chunhua Ju graduated from Xiamen University with PhD degree in 2002. Now he is a professor, doctoral supervisor and division chief of science and technology department in Zhejiang Gongshang University who focuses on intelligent information processing, data mining and collaborative innovation. And he won the award for "New Century Excellent Talents in University" of China. In the past several years, he led more than 6 national projects. He has published more than 30 papers which are SCI and EI indexed.

Chonghuan Xu received his B.S. and M.S. degrees in Computer and Information Engineering from Zhejiang Gongshang University, Hangzhou. Now he is a lecturer in College of Business Administration, ZheJiang Gongshang University. His research interests include electronic commerce, data mining. He has published over 10 publications in academic journals and conference proceedings.