A Time-Aware Weighted-SVM Model for Web Service QoS Prediction

Dou Kai, Guo Bin, and Li Kuang(✉)

School of Software, Central South University Changsha, Changsha, 410075, China
{doukai,guobin,kuangli}@csu.edu.cn

Abstract. With the rapid development of Web services, how to identify services with high Quality of Service (QoS) becomes a hot research topic. Since time-series QoS records are highly nonlinear, complex and uncertain, it is difficult to make accurate predictions through conventional mathematic methods. In order to deal with the challenging issue, this paper proposes a novel personalized QoS prediction approach considering both the temporal dynamics of QoS attributes and the influence of different QoS records. First, slide-window based data grouping is firstly utilized to obtain training dataset for regression model. Then we take the different influence of history QoS records at different time into consideration and eventually propose a weighted-SVM model for QoS prediction. Compared to Auto-Regressive Moving Average Model (ARMA), standard SVM and Collaborative Filtering (CF), the proposed approach in the paper can improve significantly the accuracy in personalized QoS prediction.

Keywords: Web service · QoS prediction · Temporal dynamics · Support vector machine

1 Introduction

Web services have become a promising technology for the development of new Internet-based Software Systems. Web services have been boosting greatly due to the strong needs of industrial companies and end users. With the rapid development of Web services, it becomes puzzling how to identify the services with high Quality of Service (QoS) from a large number of services. Web service QoS prediction is an effective way to identify the Web service which possibly has the best performance.

Influenced by the unpredictable network condition and the server workload, the Quality of Service (QoS) is dynamically changing [1] from time to time. Inaccurate Web service QoS prediction will result in large deviations in service selection. Because of the large number of services and dynamic changes of Quality of Service, it is a critical problem to be addressed in the Web services recommendation [2] and selection [3] how to utilize the historical QoS data to predict the current or future QoS value.

In this paper, we propose a novel personalized QoS prediction approach based on weighted-SVM model. By analyzing the QoS records, we find that the QoS of Web Service is affected by that in the near prior time. And the closer the interval is, the greater the influence is. Therefore, in our prediction framework, first, a slide window is proposed

S. Wang and A. Zhou (Eds.): CollaborateCom 2016, LNICST 201, pp. 302–311, 2017.
DOI: 10.1007/978-3-319-59288-6_27

for cutting the time series QoS data into multiple feature vectors as training dataset. And then, we take the different influence of history QoS records at different time into consideration through a weighted-SVM prediction model. The contributions of this paper include:

- We propose a slide to preprocess QoS historical records. Slide window is proposed for generating multiple feature vectors from one piece of time series QoS data.
- We propose a modified weighted Support Vector Machine by dynamically changing the SVM penalty factor. And we apply the weighted-SVM model to the Web services QoS prediction problem.
- By comparing with Auto-Regressive Moving Average Model (ARMA), standard SVM and Collaborative Filtering (CF), the proposed approach can improve the accuracy of personalized QoS prediction significantly.

The rest of the paper is organized as follows: Related works is discussed in Sect. 2. A Web service QoS Prediction Approach is illustrated in Sect. 3. And then the experiments and result analysis are given in Sect. 4. And finally the conclusion is drawn in Sect. 5.

2 Related Work

In the research literature, there are two mainstream technologies for QoS prediction: CF approaches and time series forecasting:

2.1 Collaborative Filtering

CF approaches are popular methods for personalized QoS prediction, which are mainly divided into two categories: model-based and neighborhood-based. Model-based approaches make predictions through learning a model. These kind of algorithms include the matrix factorization [4], the graph-based approaches [5], etc. Neighborhood-based approach utilizes historical QoS information from different users or services to measure their similarities on personalized QoS experiences. Then, QoS information from similar neighbors collected to make a personalized QoS prediction for a target user with a set of candidate Web Services. The neighborhood-based CF approaches include user-based [6], service-based [7] and hybrid [8, 9] methods. The user-based method utilizes the historical QoS experiences from similar users for personalized QoS prediction, while the service-based method uses those from similar services for prediction. The hybrid method is the combination of the previous two methods, so it can achieve higher prediction accuracy.

2.2 Time Series Forecasting

Time series forecasting approaches have been successfully applied to modeling and forecasting QoS data. They use different models to fit the past QoS values and then forecast their future changes. Godse et al. [10] proposed a method that combines monitoring technologies and extrapolation methods, which are based on ARIMA models, in order to predict future service performance. Amin et al. [11] presented an improved QoS forecasting approach which integrates ARIMA with GARCH models to address the constant variation of assumption limitation of ARIMA models. Zhang, Jinhong, Song, Jie, et al. [12] proposed a short-term prediction for QoS of web service, which are based on RBF neural networks. Zheng, Xiaoxia, et al. [13] proposed a dynamic prediction approach, which based on a time-series analysis for historical data.

Different from the above relevant work, our work is based on weighted-SVM. SVM is a machine learning method based on structural risk minimization principle [14]. Based on the analysis of QoS data, we find that different QoS records have different influence to other records. Therefore we propose a modified Support Vector Machine which has been modified by dynamically changing the SVM penalty factor, and based on the proposed weighted SVM model, we can predict the QoS based on historical records.

3 Problem Definition and Approach Overview

This section presents the problem definition and a detailed explanation of our proposed personalized QoS prediction approach.

3.1 Problem Definition

Suppose that a Web service mining system contains m users $U = \{ U_1, U_{2,} U_3 \cdots, U_m\}$ and n Web Services $S = \{ S_1, S_{2,} S_3 \cdots, S_n\}$. If user $U_i\left(U_i \in U\right)$ invoked service $S_j\left(S_j \in S\right)$ during the time interval T_k $(k = 1, 2,\cdots)$, the observed QoS value of this service invocation is recorded in the entry q_{U_i,S_jT_k} of the matrix Q. The problem we have been discussing in this paper is how to efficiently and precisely predict the future perform based on the existing records. An illustrating example of QoS records is shown in Fig. 1.

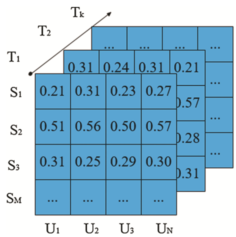

Fig. 1. Local QoS records

3.2 Theoretical Foundation

We use the dataset in WS-DREAM site [15]. The dataset consists of 142 users' invocation of 4532 Web services in 64 time slices.

We aim to find out the influence between each QoS data through statistical analysis of the dataset. Most of the QoS distributions are similar to Fig. 2. The current QoS value is close to the value at the previous and next time interval. In order to confirm whether the dataset fulfills the similar feature, we compare each QoS data with the data at next time slice by using First-order Difference function [16]. The result shows, there are 92.8% of data, the difference between the data and the next data is less than 0.1. We have a theoretical foundation that the QoS of different time periods are not independent and the QoS of Web service is affected by that of the near prior time.

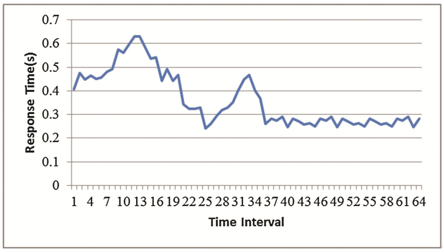

Fig. 2. Records of response-time

3.3 Approach Overview

As shown in Fig. 3, our approach mainly includes two phases: training phase and prediction phase. In the training phase, slide-window based data grouping is used to preprocess historical QoS data. Then a modified Weighted-SVM is proposed to derive the prediction model after parameter optimization. In the prediction phase, the QoS in the near future can be predicted from the derived model.

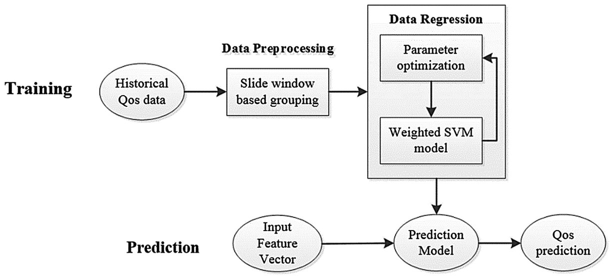

Fig. 3. Approach overview

Slide-Window based Data Grouping

In the stage of training phase, we divide each QoS sample x_i into multiple subsequences with a fixed-size slide window. A feature vector for each slide window can be generated. In order to derive a prediction model, we put the first m QoS records as training data and the size of slide window is l. We can get $m - l$ groups of mapping relationships. The training feature vectors are $(X_i, Y_i), i = (l+1, l+2, \cdots l+m)$, where $X_i = (x_{i-l}, x_{i-l-1}, \cdots, x_{i-1})$ and $Y_i = x_i$.

Figure 4 illustrates an example of feature matrix. There is a QoS time sequence S_1. And the size of slide window is l here, because there are l QoS data in each slide window. Each slide window corresponds to a training data point. Figure 4 shows two training data points (X_1, Y_1) and (X_2, Y_2). X_1 is the vector ($q_1, q_2, \cdots q_{l-1}$) and X_2 is the vector ($q_2, \cdots, q_{l-1}, q_l$). Y_1 and Y_2 represent the target value of QoS. It will cause information redundancy and introduce noise data when the size of slide window is too large. Otherwise, prediction model will get poor performance when the size of slide window is too small to contain the information we need for prediction.

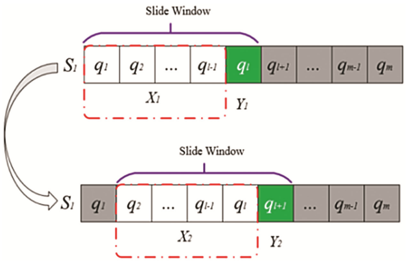

Fig. 4. Slide window

Personalized QoS Model Derivation

The basic idea of support vector regression is to map the input data X from the original input space into a high dimensional space with nonlinear map function $\emptyset(x)$. Then the problem becomes a linear regression in high dimensional space. Suppose that training samples are (x_i, y_i), $i = 1, 2, \cdots N$, where, $x_i \in R^m$ is the input vector, $y_i \in R$ is the corresponding output vector, $y = f(x)$ is the estimated output. Then we have:

$$\mathrm{y} = f(x) = \omega^T \emptyset(x) + b \tag{1}$$

where ω^T is the weight vector, $b \in R$ is the bias term. The ε-insensitive loss function is defined by:

$$L_\varepsilon(x, y, f) = |y - f(x)|_\varepsilon = max(0, |y - f(x)| - \varepsilon) \tag{2}$$

In order to obtain weight vector ω^T and bias term b, we must minimize the sum of the ε-insensitive loss function:

$$\frac{1}{2}||\omega^2|| + C\sum\nolimits_{i=1}^{N} L_\varepsilon\left(x_i, y_i, f\right) \tag{3}$$

where parameter C can measure the trade-off between complexity and losses. We can convert formula (3) into an equivalent formula (4) by introducing slack variables ξ_i and ξ_i^*.

$$\min[\frac{1}{2}||\omega||^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*)]$$
$$\text{s.}t.\begin{cases} y_i - \left[\omega^T\emptyset(x) + b\right] \leq \varepsilon + \xi_i \\ \left[\omega^T\emptyset(x) + b\right] - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{4}$$

Because of the high dimensional feature space, it is not easy to find the answer of formula (4). We can convert it into a dual problem by introducing kernel function:

$$\max j\left(\alpha_j, \alpha_i^*\right) = \sum_{i=1}^{\mathrm{N}} \left(\alpha_i^* - \alpha_i\right)\mathrm{y}_i - \varepsilon\sum_{i=1}^{\mathrm{N}} \left(\alpha_i^* + \alpha_i\right) - \frac{1}{2}\sum_{i,j=1}^{\mathrm{N}} \left(\alpha_i^* - \alpha_i\right)\left(\alpha_j^* - \alpha_j\right)\mathrm{K}\left(x_j, x\right)s.t.\begin{cases} \sum_{i=1}^{N}\left(\alpha_i - \alpha_i^*\right) = 0 \\ \alpha_i - \alpha_i^* \in [0, C] \end{cases} \tag{5}$$

where α_i, α_i^* are the corresponding Lagrange multipliers. And there are some kernel functions such as Linear kernel, Polynomial kernel, Sigmoid Kernel and RBF Kernel.

In the standard SVM, all the samples with deviation over ε have the same penalty factor C. Because of the different influence of various QoS samples, standard SVM have a poor performance on QoS prediction. To solve this problem, we give different penalty factor to the samples with deviation over ε. The closer the time is, the greater the penalty factor C is. The weighted function $W(i)$ can improve the accuracy of the prediction model. Therefore we modify formula (3) to (6).

$$\frac{1}{2}||\omega^2|| + C\sum_{i=1}^{N} W(i) * L_\varepsilon\left(x_i, y_i, f\right) \tag{6}$$

And we can modify formula (5) to formula (7):

$$maxJ\left(\alpha_i, \alpha_i^*\right) = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)y_i - \varepsilon\sum_{i=1}^{N}\left(\alpha_i^* + \alpha_i\right) - \frac{1}{2}\sum_{i,j=1}^{N}\left(\alpha_i^* - \alpha_i\right)\left(\alpha_j^* - \alpha_j\right)K\left(x_i, x\right)$$
$$\text{s.}t.\begin{cases} \sum_{i=1}^{N}(\alpha_i - \alpha_i^*) = 0 \\ \alpha_i, \alpha_i^* \in [0, C * W(i)] \end{cases} \tag{7}$$

We choose exponential function as weighting function from linear function, quadratic parabolic function and exponential function based on extensive experiments:

$$W(i) = 1/(1 + \exp(-i))(i = 1, 2, \cdots l) \tag{8}$$

Where $i = 1$ represents the first QoS sample of training data, $i = l$ represents the last QoS sample of training data. $W(i)$ is a monotonically increasing function. Hsu [17] points out that RBF kernel has a good performance in most instances based on many years of research and analysis. So we select RBF kernel as kernel function.

$$K(x, y) = exp\left(\frac{-||x - x_i||^2}{\sigma^2}\right) \tag{9}$$

By solving the above quadratic programming, we can obtain α_i and α_i^*. Through the expression (11), we can get our regression function.

$$\hat{Y} = \sum_{i=1}^{N} \left(\alpha_i - \alpha_i^*\right) exp\left(\frac{-\|x - x_i\|^2}{\sigma^2}\right) + b \tag{10}$$

The weighted-SVM model is proposed here for Web services QoS prediction. SVM model can map the original nonlinear data in a low dimensional space into that with linear feature in a high dimensional space. And the kernel function is used to solve the optimization problem of the loss function. In the training phase, we aim to get the appropriate C,ε, σ^2 for our prediction model. Finally, QoS values can be predicted based on the trained weighted SVM model.

4 Implementation and Experiment

4.1 Dataset Collection

The experiment is implemented in a dataset from WS-DREAM site. In order to analyze the property of our data, we count the scale of the data. We find QoS data lies between 0 s and 20 s. The mean of these data is 2.328 s (Table 1).

Table 1. Information of data

Statistics	Response-time
Scale	0–20 s
Mean	2.328 s
Num. of users	142
Num. of service	4532
Num. of time slices	64

4.2 Evaluation Criterion

Mean Absolute Error (MAE) is used here for model evaluation. The definition of MAE is:

$$MAE = \frac{\sum_{ijk} |Y_{ijk}^{\wedge} - Y_{ijk}|}{N} \tag{11}$$

In this formula, $Y_{ijk}^{\wedge}$ is the predicted value of the response time that user i invocate service j at time k. Y_{ijk} is the real value under the same condition. N is the number of QoS that we need to predict.

4.3 Analysis of Parameter

In our experiment, we aim to find appropriate C,ε,σ^2 and the size of slide window l. We divide the parameters adjustment experiment into four parts. First we adjust ε from 0.01 to 0.1. And then we adjust the parameter of σ^2 from 0.1 to 0.9 in the second part. Then we change C from 10 to 100. At last we change the size of the slide window from 1 to 8. Take the records of U_1 invoking services S_1 as an example, Fig. 6 shows that ε, σ^2, C, l are 0.03, 0.6, 30, 4 are the optimal values for the QoS prediction model of user U_1 invoking services S_1. The result is shown in Fig. 5

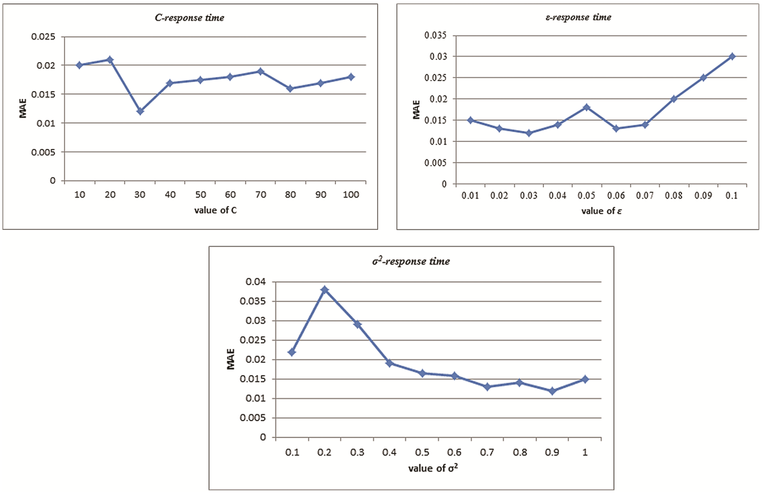

Fig. 5. Impact of C, ε, σ^2

In order to choose the size of slide window, we select 100 users and 50 different Web services as experimental data. We can see from Fig. 6 that the MAE of different QoS series is leveling off when the size of slide window is 5. So we choose $l = 5$.

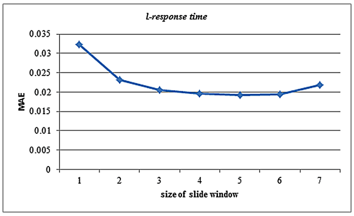

Fig. 6. Impact of l

4.4 Performance Comparisons

We choose the 100 users invoking the first 50 services records as the experimental data. And the size of slide window is 5. The former 54 feature vectors are selected as training data and the prediction is made on the latter 10 QoS values. We compare the prediction accuracy of the following methods:

- CF-This is a tensor factorization-based prediction method [12]. It applies tensor factorization on user-service-time tensor to extract user-specific, service-specific and time-specific characterizes.
- ARMA-ARMA is originally proposed by Box and Jenkins [18] and is a very popular model for time series forecasting.
- Standard SVM-SVM [14] is a novel machine learning method based on structural risk minimization principle.
- Weighted-SVM-This method is proposed in this paper. It is a modified SVM with dynamic penalty factor

The results can be seen from Fig. 7, the performance of CF is worse than other methods. Because prediction based on Collaborative Filtering only extracts the user-specific and service-specific features without considering the relationship between QoS changes in time intervals. Comparison between ARMA model and SVM prediction model, ARMA model shows a good performance when the QoS data are stable. But most of QoS record is highly nonlinear, complex and uncertain, it is difficult to make accurate predictions using ARIMA model. SVM have a better performance while solving nonlinear QoS regression problem. Comparison between standard SVM model and the proposed weighted-SVM model, standard SVM has a poor performance of QoS prediction because standard SVM cannot reflect the different influence between different QoS samples with a same penalty factor *C*.

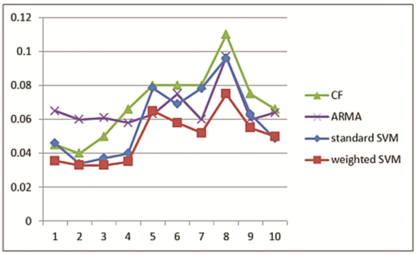

Fig. 7. Performance comparison

5 Conclusion

In this paper, we proposed a Web service QoS prediction framework based on weighted-SVM model. Firstly, we use a slide window for data modeling. Then a weighted-SVM model is employed to predict the QoS of Web services. In regard of practical application, our model has great space to be improved. Although it has achieved better accuracy compared to ARMA, CF and standard SVM, more improvement is needed in practical using. From the view of current stage, the combination of time and space is remarkable. But other ways of combination is worth to try to make a better model.

Acknowledgement. The research is supported by "Natural Science Foundation of Hunan Province" (No.2016JJ3154), "National Natural Science Foundation of China" (No.61202095), "Scientific Research Project for Professors in Central South University, China" (No. 904010001), and "Innovation Project for Graduate Students in Central South University" (No. 502210017).

References

1. Haner, M., Hinton, D., Klein, T.E.: Method of dynamically adjusting quality of service (QoS) targets: US, US 20060212594 A1[P] (2006)
2. Alrifai, M., Risse, T.: Combining global optimization with local selection for efficient QoS-aware service composition. In: Proceedings of the 18th International Conference on World Wide Web (2009)
3. Zheng, Z., Ma, H., Lyu, M.R., et al.: QoS-aware web service recommendation by collaborative filtering. IEEE Trans. Serv. Comput. **4**(2), 140–152 (2011)
4. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. Nips **1**(1), 1257–1264 (2012)
5. Fouss, F., Pirotte, A., Renders, J.M., et al.: Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. IEEE Trans. Knowl. Data Eng. **19**(3), 355–369 (2007)
6. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings 14th Conference on Uncertainty Artificial Intelligence, pp. 43–52 (1998)
7. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inf. Syst. **22**(1), 143–177 (2004)
8. Zheng, Z., Ma, H., Lyu, M.R., King, I.: WSRec: A collaborative filtering based Web service recommender system. In: ICWS, pp. 437–444 (2009)
9. Zheng, Z., Ma, H., Lyu, M.R., King, I.: QoS-aware Web service recommendation by collaborative filtering. IEEE Trans. Serv. Comput. **4**(2), 140–152 (2011)
10. Godse, M., Bellur, U., Sonar, R.: Automating QoS based service selection. In: ICWS, pp. 534–541 (2010)
11. Amin, A., Colman, A., Grunske, L.: An approach to forecasting QoS attributes of web services based on ARIMA and GARCH models. In: ICWS, pp. 74–81 (2012)
12. Zhang, J., Song, J.: A short-term prediction in web service QoS based on RBF neural network. J. Liaoning Eng. Technol. Univ. **29**(5), 918–921 (2010)
13. Zheng, X., Zhao, J., Cheng, Z., Xie, B.: A WebService response time dynamic prediction method. In: Mini-micro Systems, vol. 8 (2011)
14. Yang, J.F., Zhai, Y.J., Wang, D.F., et al.: Time series prediction based on support vector regression. Proc. CSEE **25**(17), 110–114 (2005)
15. Zhang, Y., Zheng, Z., Lyu, M.R.: WSPred: A time-aware personalized QoS prediction framework for web services. In: Proceedings of the 22th IEEE Symposium on Software Reliability Engineering (ISSRE 2011) (2011)
16. Kurogi, S., Koyama, R., Tanaka, S., et al.: Forecasting using first-order difference of time series and bagging of competitive associative nets. In: 2007 International Joint Conference on Neural Networks, IJCNN 2007, pp. 166–171. IEEE (2007)
17. Hsu, B.C., Chang, C.-C., Lin, C.-J.: A practical guide to support vector classification (2012)
18. Box, G.E., Jenkins, G.M.: Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco (1976). revised ed.