An SVR-Based Data Farming Technique for Web Application

Jian Lin[1] and Minjing Peng[2]

[1]School of Economics and Management, Beihang University 100083 Beijing, P.R. China
Jianlin@wyu.cn
[2] Institute of Systems Science and Technology, Wuyi University, Jiangmen 529020, Guangdong, P.R. China reggiepeng@163.com

Abstract. In order to solve the problem that the performance of web application can't meet users' expectation when many users access dynamic data through the application, a data farming optimizing technique for web application is proposed. In the proposed technique, the web application is based on XML and the data are converted into XML format before being presented through XSL. To reduce page response time, SVR is employed to forecast user's requests. And based on the requests, a data farming agent is used to create web pages that may be needed by users. At last, experiments are conducted to compare page response time for three web sites: a dynamic web site without data farming optimization, a data farming optimized web site, and a SVR based data farming optimized web site. It is proved that page response time for the last web site is the least, which proves that the proposed technique is effective.

Keywords: *Data farming, Web application, SVR, XML, Agent*

1. INTRODUCTION

In the e-commerce era, dynamic information customized for each user in web applications provides business the opportunity of success. And performance of web applications is a critical factor to the success of the business. However, the dynamic information is presented from dynamic data in database through dynamic web pages, which require much processing time, and hence the process would lower the performance when there are many users accessing the web application. At this time, opportunities are that the operating systems are over-worked. There are two ways to solve the problem. One solution is upgrading the hardware supporting web applications. However, the upgrade requires more expenditure and more maintenance efforts. The other solution is utilizing the processing capability in the idle time through the technique of data farming. And it is also proposed in this paper. Data farming means that the processing ability of the idle time is used to generate static web pages that are possibly be used in next operational session. In the proposed technique, Support Vector Regression, or SVR, is employed to forecast users' needs so that the appropriate static pages can be prepared.

The rest of this paper is organized as follows. In section 2, concepts of web application are introduced. And as an important measure of performance, response

time is described and analyzed. In section 3, concepts, strengths and applications of data farming are presented. In section 4, fundamentals of SVR are shown. And then, in section 5, the framework of the proposed technique is illustrated. And in section 6, experiments of three web sites: a dynamic web site without data farming optimization, a data farming optimized web site, and a SVR based data farming optimized web site are conducted to prove that the proposed technique is effective.

2. WEB APPLICATION

A web application is a multi-tier application system, which communicates with an application server through Hypertext Transfer Protocol (HTTP) and Transmission Control Protocol / Internet Protocol (TCP/IP). It is capable of dynamically processing data and then presenting the information to users through web browsers.

Strengths of web applications are: a) User-friendly interface. All information of web application is presented through web browsers, which makes the application easy to use and the users don't need specialized training; b) Easy maintenance and upgrades. Because that there is no need to install specialized software in the client side, the only job needed to upgrade the system is upgrading the software system in the server side, and it reduces maintenance and upgrade costs; c) Scalability. The use of standard TCP and HTTP enables the module independent and thus makes system easy to expand; and d) High degree of information sharing. Web applications use HTML/XML to transmit information, which means the data format is an open standard. And hence the industry has given web application broad supports.

Due to above strengths, web applications have been widely used to provide customers and partners with customized, convenient, cheap, and efficient information services. And thus, competitiveness of enterprises is promoted. However, large accesses hinder web applications from meeting user performance expectation, which would consequently hinder the success of business.

The performance is measured with response time [1]. Response time of Web applications is the time spent in a HTTP request. It starts when the client sends a HTTP request and ends when the client receives the first byte data from the server. Response time under multi-user requests refers to the response time for a user who randomly accesses the websites with some other users. The response time under multi-user requests is denoted by τ_n^i , where superscript i refers to the user identification, and subscript n refers the total users. Assume that the response times for n users who concurrently access the web application are $\tau_n^1,\cdots,\tau_n^i,\cdots,\tau_n^n$, the average response time $\overline{\tau}_n$ could be computed using equation (1).

$$\overline{\tau}_n = \frac{\tau_n^1 + \cdots + \tau_n^i + \cdots, \tau_n^n}{n} \tag{1}$$

3. DATA FARMING

Data farming maximizes system performance by ensuring that information delivered to users by web application is available when the users require them. The data farming process is executed by a data farming agent, which are called as a data farmer. It is an independent computer process that is constantly looking for opportunities to improve the ability of web application to serve customers. It watches the web application server performance statistics and starts framing at any point in which the demand on the server drops below a defined threshold. And the process consists of four steps: a) Select seeds, b) Grow, c) Harvest and d) Store. The process is illustrated in figure 1.

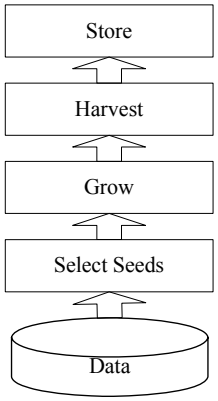

Figure 1. **The Process of Data Farming**

3.1 Select Seeds

Achieving largest profits by providing right products or services to customers is the purpose of e-commerce web application. And what products or services customers will choose could be predicted through historical transaction records [2, 3]. This research focuses on the data of enterprise customers and profitable products these customers may purchase, and takes them as seeds and grows them into prepared web pages presenting information to customers.

Seed is a combination of features from users and products or services that are to be delivered by a web application. In this step, seeds that are to be nurtured are selected according to the forecasting results based on features of users and products or services using SVR. The criterion for choosing is the probability of users may interest in the services or products.

3.2 Grow

In this step, selected seeds need to grow into static web pages. These seeds and some relevant data will be collected together to form the information that users need. The relevant data are classified into three classes: relevant data of customers, relevant data of products, and system information. For example, product recommendation page needs user's information of login status, information about products and user's profile. And the information from these sources is collected in this step.

3.3 Harvest

The collected data are structured using XML format and transformed into web pages via XSL [4]. XML stands for eXtensible Markup Language. It was designed to describe data and to focus on what data is. And XSL is a family of recommendations for defining XML document transformation and presentation. The process of harvest is illustrated in figure 2.

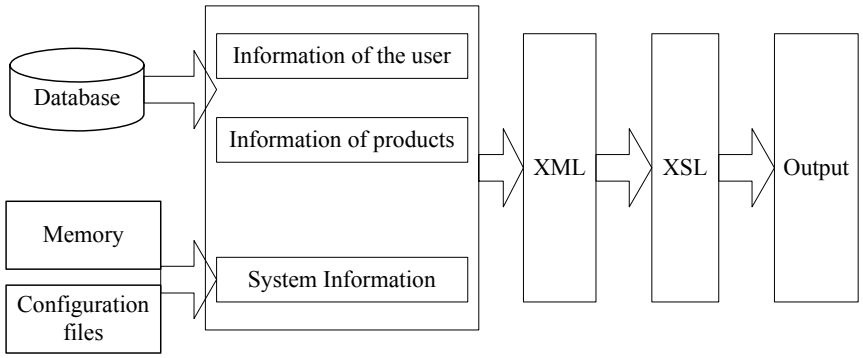

Figure 2. **The Process of Harvest**

The process fully exploits the benefits of XML through its 'virtual XML tree' approach. Each output pages from the system derives from an XML source.

3.4 Store

These pages are stored in somewhere in file system of web server, so that they can be accessed when users request. And those frequently accessed pages are cached to improve the performance of the application.

4. SVR

In this technique, SVR is employed to forecast the probability of a customer purchasing a product. According to the probability, seeds are selected to grow into static web pages to reduce workload of application in busy time.

Statistical Learning Theory proposed by Vapnik is a theory specialized on the learning problem through a limited number of observations [5]. Support Vector Machine (SVM) proposed based on the theory is a new way for solving nonlinear problems. Through introducingε- insensitive loss function, SVM has been extended to solve problems of nonlinear regression. And the new technique is called as Support Vector Regression.

For a sample dataset $\{(x_1, y_1), \cdots, (x_l, y_l)\} \subset X \times \mathbb{R}$, the purpose of linear regression is to find the weight coefficient w in equation (2).

$$y = Xw + \sigma \tag{2}$$

Where, σ is a coefficient to be minimized. Considering the purpose of regression function and ε -insensitive loss function, linear regression problem can be transformed into following optimizing problem:

$$\min \Phi(w, \xi^*, \xi) = \frac{1}{2}(w \cdot w) + C\sum_{i=1}^{l}\left(\xi_i + \xi_i^*\right) \tag{3}$$

Where, l is the number of observation, C is a fixed value that is used to split the regression error and function feature. Above equation can be transformed into the following optimizing problem:

$$\begin{aligned} \max W(\alpha, \alpha^*) = & \sum_{i=1}^{l}\left(\alpha_i^*(y_i - \varepsilon) - \alpha_i(y_i + \varepsilon)\right) \\ & -\frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i^* - \alpha_i\right)\left(\alpha_j^* - \alpha_j\right)\left(x_i \cdot x_j\right) \end{aligned} \tag{4}$$

$$s.t.\begin{cases} 0 \le \alpha_i \le C, i = 1, 2, \cdots, l \\ 0 \le \alpha_i^* \le C, i = 1, 2, \cdots, l \\ \sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0 \end{cases} \tag{5}$$

For nonlinear regression, the kernel function is introduced to replace the inner product $\left(x_i \cdot x_j\right)$. And the optimizing problem in equation (4) could be transformed into the following equation.

$$\max W(\alpha,\alpha^*) = \sum_{i=1}^{l}\left(\alpha_i^*(y_i-\varepsilon)-\alpha_i(y_i+\varepsilon)\right) - \frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i^*-\alpha_i\right)\left(\alpha_j^*-\alpha_j\right)K\left(x_i,x_j\right) \quad (6)$$

Constraints in equation (6) are the same as in equation (5). Values of parameters α_i and α_i^* can be obtained by solving equation (6). On the basis of KKT conditions, there are not many α_i and α_i^* that are unequal to 0. Samples (X_i, y_i) corresponding to α_i and α_i^* are referred as support vectors. The regression function is in the following form:

$$f(x) = \sum_{SVs}(\alpha_i - \alpha_i^*)K(x_i, x) + b \quad (7)$$

Where,

$$b = -\frac{1}{2}\sum_{SVs}(\alpha_i - \alpha_i^*)[K(x_r, x_i) + K(x_s, x_i)] \quad (8)$$

And x_r, x_s are support vectors. Equation (7) is the SVR model. From above discussion, we can find that it has two advantages. First, it is based on kernel mapping which ensures its nonlinear processing ability. Second, it is based on statistical learning theory, so it synthetically takes fitting error and function characteristics of regression model into account, so that its generalization ability is also guaranteed.

5. SVR BASED DATA FARMING TECHNIQUE

5.1 Framework

The whole application can be in two phases: busy time phase and idle time phase.

In the busy time phase, the process of application is busy with the requests from web users. When the process can't find a corresponding static web page for a request, the job of response would be completed by dynamic pages. And the only job data farmer does is watching CPU, when the percentage of idle time is larger than a predefined a threshold, the application gets into idle time phase [6].

In the idle time phase, data farmer needs to do a job besides watching CPU: farming data of customers and products, and generating static web pages. When the percentage of idle time is below the predefined threshold, the process goes into the busy time phase again.

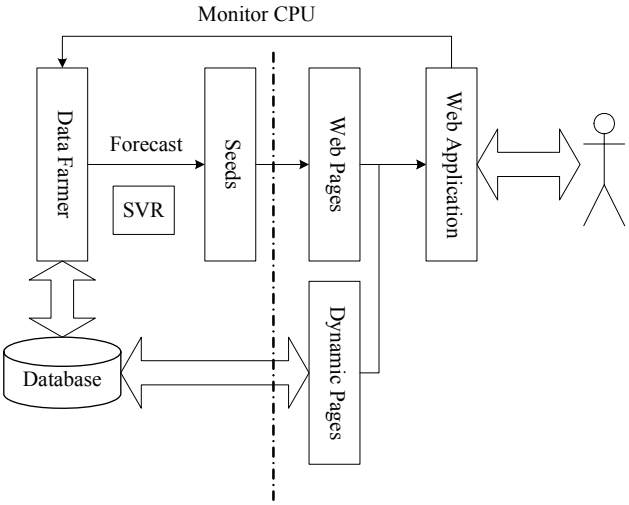

Figure 3. Framework

5.2 Implementation

The database includes historical database and the customer database. All customer transactions records are stored in historical memory database, and all customers' profiles are stored in customer database. Database is based on Microsoft SQL Server 2000 platform.

As SVR is in MatLab code [7], and application server is Microsoft .Net 2005, Combuilder of MatLab is employed to compile SVR MatLab code into COM dll [8], and then SVR can be called by .NET 2005.

Data farmer is implemented as an independent process, which is responsible for monitoring CPU and generating static web pages.

6. EXPERIMENTS

A website without data farming optimization, a data farming optimized web site, and a SVR based data farming optimized web site are built to verify the performance of a web application. The web application is a Blog web site and provides free basic service. However, its extra services such as theme services charge.

The services are provided by a PC server with 2G memory. And the operating system is windows 2003 server standard version. Experiment parameters are shown in table 1.

Table 1. Experiment Parameters

Parameter	Value
Records for training	317
SVR kernel function	$\exp\left(-\left\|x_i - x_j\right\|^2 / \sigma^2\right)$
SVR insensitive loss function ε	0.1
SVR punishment factor C	60
σ^2	6

LoadRunner 8.0 is used to conduct the test [9]. And the range of the number of concurrent users is from 50 to 500. Experiment results are plotted in figure 4.

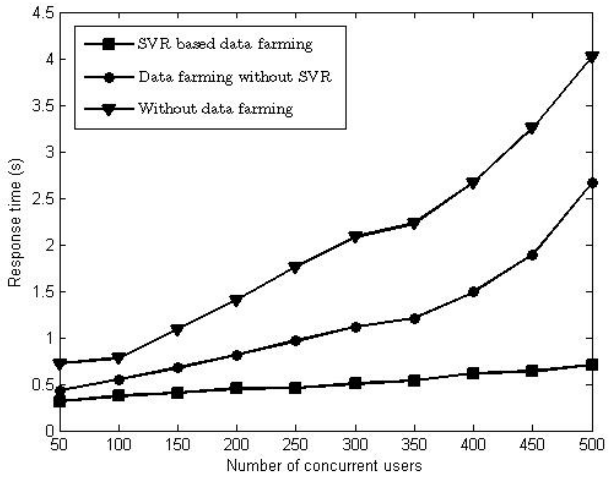

Figure 4. A Comparison of Experimental Results

The results indicate that data farming is an effective technique for improving performance of web application in busy time by transforming dynamic web pages into static web pages in the idle time. And the proposed SVR based data farming technique performs best in that SVR helps select suitable seeds and hence improve the efficiency of data farming.

7. CONCLUSIONS

Data farming is an effective technique for improving performance of web application in busy time by transforming dynamic web pages into static web pages in the idle time. Through the forecast of SVR, the proposed technique performs well in improving efficiency of web application.

REFERENCES

1. Anonymous, *Response Time,* TechTarget (2007). http://searchnetworking.techtarget.com/sDefinition/0,sid7_gci212896,00.html (Accessed July 14, 2007).
2. D.C. Schmittlein, D.G. Morrison, and R. Colombo, Counting your customers: who are they and what will they do next, *Management Science.* Volume 33, Number 1, pp.1-24, (1987).
3. D.C. Schmittlein and R.A. Peterson, Customer base analysis: an industrial purchase process application, *Marketing Science.* Volume 13, Number 1, pp.41-67, (1994).
4. Anonymous, *Extensible Markup Language*, W3C (2007). http://www.w3.org/xml (Accessed July 14, 2007).
5. V.N. Vapnik, *Statistical Learning Theory* (Wiley: New York, 1998).
6. D. Burnell, A. Al-Zobaidie, G. Windall, and A. Butler, Self-Optimising Data Farming for Web Applications, in *Proc. of the 15th International Workshop on Database and Expert Systems Applications (DEXA 2004)*, eds. F. Galindo, M. Takizawa, and R. Traunmüller (Springer: Boston, MA 2004), pp.436-440.
7. G.C. Cawley, *MATLAB Support Vector Machine Toolbox (v0.55\beta)*, School of Information Systems, University of East Anglia, U.K. (2000). http://Theoval.sys.uea.ac.uk/svm/toolbox/ (Accessed July 14, 2007).
8. Anonymous, *Create MATLAB based .NET and COM components*, Mathworks (2006). http://www.mathworks.com/products/netbuilder/ (Accessed July 14, 2007).
9. Anonymous, *Load Testing Software: Automated Performance Testing and Web Testing Software*, Mercury (2007). http://www.mercury.com/us/products/performance-center/loadrunner/