Journal of Intelligent Transportation Systems: Technology, Planning, and Operations
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/gits20

Bus Arrival Time Prediction Using Support Vector Machines

Yu Bin [a] , Yang Zhongzhen [a] & Yao Baozhen [a]
[a] Transportation College , Dalian Maritime University , Dalian, China
Published online: 22 Oct 2007.

To cite this article: Yu Bin , Yang Zhongzhen & Yao Baozhen (2006) Bus Arrival Time Prediction Using Support Vector Machines, Journal of Intelligent Transportation Systems: Technology, Planning, and Operations, 10:4, 151-158

To link to this article: http://dx.doi.org/10.1080/15472450600981009

Journal of Intelligent Transportation Systems, 10(4):151–158, 2006

ISSN: 1547-2450 print / 1547-2442 online
DOI: 10.1080/15472450600981009

Bus Arrival Time Prediction Using Support Vector Machines

YU BIN, YANG ZHONGZHEN, and YAO BAOZHEN

Transportation College, Dalian Maritime University, Dalian, China

Effective prediction of bus arrival time is central to many advanced traveler information systems. This article presents support vector machines (SVM), a new neural network algorithm, to predict bus arrival time. The objective of this paper is to examine the feasibility and applicability of SVM in vehicle travel time forecasting area. Segment, the travel time of current segment, and the latest travel time of next segment are taken as three input features. Bus arrival time predicted by the SVM is assessed with the data of transit route number 4 in Dalian economic and technological development zone in China and conclusions are drawn.

Keywords Prediction; Bus Arrival Time; Support Vector Machine

Within the context of intelligent transportation systems, advanced public transportation systems (APTS) and advanced traveler information systems (ATIS) are designed to collect, process, and disseminate information to transit users via emerging navigation and communication technologies (Federal Transit Administration 1998). One of the key elements in APTS/ATIS is a model to predict transit vehicle arrival time with reasonable accuracy. The arrival time deviations of buses are usually caused by several stochastic factors (traffic congestion, ridership distribution, and weather condition). The resulting impact of these factors on the transit system comprises bunching between pairs of operating vehicles, increasing passengers waiting time, deterioration of schedule/headway adherence, unsmooth intermodal transfers, increasing the cost of operation, traffic delays, etc. All these factors will reduce the level of service of the transit agency and discourage riders from using the transit system. One way to mitigate the impacts is to provide accurate information of vehicle arrival/departure time and expected delays at all major stops.

Transit vehicle arrivals at stations/stops in urban networks are stochastic because travel time on links, dwell time at stops, and delays at intersections fluctuate spatially and temporally. Thus, developing such a model that can adapt to time varying traffic and demand conditions is a challenging task. Existing literatures focused on three types of models: time series, artificial neural network, and Kalman filtering technique. The time series models can obtain an understanding of the underlying forces and structure that produced the observed data. But the accuracy of time series models highly relies on the similarity between the real-time and historical traffic patterns. The variation of the historical average could cause significant inaccuracy in the prediction results (Smith and Demetsky 1995). And these methods usually have a short time lag while predicting in real time (Stephanedes et al. 1990; DeLurgio 1998). Originating from the state-space representations in modern control theory, Kalman filtering models have been applied for predicting short-term traffic demand and travel time on freeways (Okutani and Stephanedes 1984; Stephanedes et al. 1990; Dailey et al. 2001; Shalaby and Farhan 2003). Kalman filtering models have elegant mathematical representations (e.g., linear state-space equations) and the potential to adequately accommodate traffic fluctuations with their time-dependent parameters (e.g., Kalman gain). These models were effective for predicting the travel time one time step ahead, but they deteriorated when the forecasting had to be done over multiple time steps (Park and Rilett 1999). Artificial neural networks (ANNs), motivated by emulating the intelligent data processing ability of human brains, are constructed with multiple layers of processing units named artificial neurons. Meanwhile, the synaptic weights can be adjusted to map the input-output relationship for the analyzed system automatically through a learning process (Hagan et al. 1996; Wei and Wu 1997). With such versatile parallel distributed structures and adaptive learning processes ANNs appear to be a promising approach to describe complex systems where various time and location dependent factors are intercorrelated. Also, ANNs have been recently

Address correspondence to Yu Bin, Transportation College, Dalian Maritime University, Dalian, 116026, P. R. China. E-mail: minlfish@yahoo.com.cn

gaining popularity in vehicle travel time prediction (Ding and Chien 2000; Chien et al. 2002; Chen et al. 2004). However, it has been commonly reported that ANN models require a large amount of training data to estimate the distribution of input pattern and they have difficulty generalizing the results because of their overfitting nature. In addition, it fully depends on researchers' experience or knowledge to preprocess data in order to select control parameters including relevant input variables, hidden layer size, learning rate, and momentum (Lawrence et al. 1997; Moody 1992; Sarle 1995; Weigend 1994).

Recently, SVM has been proposed as a novel technique in time series forecasting (Mukherjee et al. 1977; Muller et al. 1997, 1999). Although SVM also depends on the similarity between historic and real-time traffic patterns, it still shows many breakthroughs and plausible performance, such as forecasting of financial market (Yang et al. 2002) and forecasting of electricity price (Sansom et al. 2002). For intelligent transportation systems (ITSs), there also are many researches, such as vehicle detection (Sun et al. 2002), traffic-pattern recognition (Ren et al. 2002), head recognition (Reyna et al. 2001), and travel time prediction of highway (Wu et al. 2004). These research results evidence the feasibility of SVM in ITS.

SVM is a very specific type of learning algorithm characterized by the capacity control of the decision function, the use of the kernel functions, and the sparsity of the solution (Cristianini and Taylor 2000; Vapnik 1999, 2000). Established on the unique theory of the structural risk minimization principle to estimate a function by minimizing an upper bound of the generalization error, SVM is shown to be very resistant to the overfitting problem, eventually achieving high generalization performance in solving various time series forecasting problems (Cao and Tay 2001; Tay and Cao 2001). Another key property of SVM is that training SVM is equivalent to solving a linearly constrained quadratic programming problem so that the solution of SVM is always unique and globally optimal, unlike other networks' training which requires nonlinear optimization with the danger of getting stuck into local minima.

In order to predict travel time in an accurate and timely manner the consideration of traffic condition is essential, including traffic congestion, etc. Because traffic congestion is complicated and difficult to use, in this research, the travel time of preceding/current bus on links is used to consider traffic conditions of links. The objectives of this research are to develop and apply SVM models to predict the arrival time. This article is organized as follows: the first section provides a brief introduction to SVM regression; the next contains results and analyses including performance evaluation of the methodology; and lastly, the conclusions are presented.

THEORY OF SVM FOR REGRESSION

SVMs are based on the structural risk minimization (SRM) inductive principle, which seeks to minimize an upper bound of the generalization error consisting of the sum of the training error and a confidence level. This is the difference from commonly used empirical risk minimization (ERM) principle, which only minimizes the training error. Based on such induction principle, SVM usually achieves higher generalization performance than the traditional neural networks that implement the ERM principle in solving many machine learning problems. Another key characteristic of SVM is that training SVM is equivalent to solving a linearly constrained quadratic programming problem so that the solution of SVM is always unique and globally optimal, unlike other network's training which requires nonlinear optimization with the danger of getting stuck in local minima. In SVM the solution to the problem is only dependent on a subset of training data points which are referred to as support vectors. Using only support vectors, the same solution can be obtained as using all the training data points. One disadvantage of SVM is that the training time scales somewhere between quadratic and cubic with respect to the number of training samples. So a large amount of computation time will be involved when SVM is applied for solving large-size problems (Cao et al., 2003). However, in this study, a small data pool is considered.

Given a set of data points $(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)$ (x_i is an n-dimensional input vector, such as segment and travel time of current segment, etc. $x_i \in X \subseteq R^n$, X denotes the input vector space; y_i is the desired value, such as observed travel time of next segment. $y_i \in Y \subseteq R$, l is the number of training samples, Y denotes the output variable space) are randomly and independently generated from an unknown function. SVM approximates the function using the following form (Cao et al., 2003):

$$f(x) = \omega \bullet \phi(x) + b \tag{1}$$

where $\phi(x)$ represents the high-dimensional feature spaces which are nonlinearly mapped from the input space x. The coefficients ω and b are estimated by minimizing the regularized risk function:

$$\frac{1}{2}\|\omega\|^2 + C\frac{1}{l}\sum_{i=1}^{l} L_\varepsilon(y_i, f(x_i)) \tag{2}$$

The first term $\|\omega\|^2$is called the regularized term. Minimizing $\|\omega\|^2$will make a function as flat as possible, thus playing role of controlling the function capacity. The second term $\frac{1}{l}\sum_{i=1}^{l} L_\varepsilon(y_i, f(x_i))$ is the empirical error measured by the ε-insensitive loss function, which is defined below (Vapnik 2000):

$$L_\varepsilon(y_i, f(x_i)) = \begin{cases} |y_i - f(x_i)| - \varepsilon, & |y_i - f(x_i)| \geq \varepsilon \\ 0 & \mathit{otherwise} \end{cases} \tag{3}$$

This defines aε tube (Figure 1) so that if the predicted value is within the tube the loss is zero, while if the predicted point is outside the tube, the loss is the magnitude of the difference between the predicted value and radius ε of the tube. C is called the regularization constant. Increasing the value of C will result in the relative importance of the empirical risk with respect to the regularization term to grow. ε is called the tube size and it is

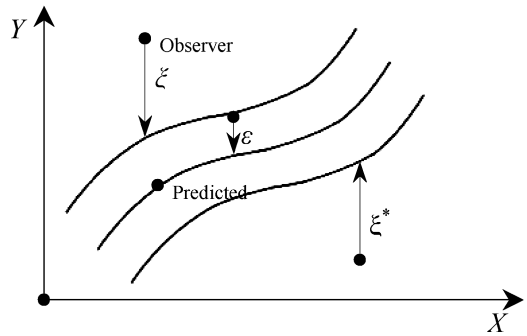

Figure 1 The parameters for the support vector regression.

equivalent to the approximation accuracy placed on the training data points. Both C and ε are user-prescribed parameters.

We assume that there is a function $f\ (\ )$ that approximates training data x with precision ε. In this case we assume that the problem is feasible. In the case of infeasibility, one can introduce slack variables ξ_i, ξ_i^* to cope with infeasible constraints of the optimization problem. Then the above problem can be formalized as:

$$\min \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \tag{4}$$

Subject to

$$y_i - \omega \cdot \phi(x_i) - b \le \varepsilon + \xi_i \tag{5}$$

$$\omega \cdot \phi(x_i) + b - y_i \le \varepsilon + \xi_i^*, i = 1, \ldots, l \tag{6}$$

$$\xi_i^* \ge 0 \tag{7}$$

In most cases the optimization problem (4) can be solved more easily in its dual Eq. (8).

$$L = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) - \sum_{i=1}^{l}(\eta_i\xi_i + \eta_i^*\xi_i^*)$$

$$-\sum_{i=1}^{l} a_i(\varepsilon + \xi_i - y_i + \omega \bullet \phi(x) + b)$$

$$-\sum_{i=1}^{i} a_i^*(\varepsilon + \xi_i^* - y_i - \omega \bullet \phi(x_i) + b) \tag{8}$$

Here L is the Lagrangian and $\eta_i, \eta_i^*, a_i, a_i^*$ are Lagrange multipliers. Hence, the dual variables in (8) have to satisfy positive constraints.

$$\eta_i^*, a_i^* \ge 0 \tag{9}$$

Again, we refer to a_i, a_i^*.

It follows from the saddle point condition that the partial derivatives of L with respect to the primal variables.

$$\partial_b L = \sum_{i=1}^{l}(a_i + a_i^*) = 0 \tag{10}$$

$$\partial_\omega L = \omega - \sum_{i=1}^{l}(a_i + a_i^*)\phi(x_i) = 0 \tag{11}$$

$$\partial_{\xi_i^*} L = C - a_i^* - \eta^* = 0 \tag{12}$$

Substituting (10), (11), and (12) in (8) yields the dual optimization problem.

$$\max W(a_i, a_i^*) = \sum_{i=1}^{l} y_i(a_i - a_i^*) - \varepsilon\sum_{i=1}^{l}(a_i + a_i^*)$$

$$-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(a_i - a_i^*)(a_j - a_j^*)(\phi(x_i) \bullet \phi(x_j)) \tag{13}$$

$$\text{Subject to } \sum_{i=1}^{l}(a_i - a_i^*) = 0 \quad \text{and } a_i, a_i^* \in [0, C]; \tag{14}$$

Through condition (11) we can get

$$\omega - \sum_{i=1}^{l}(a_i - a_i^*)x_i = 0 \tag{15}$$

thus

$$f(x) = \sum_{i=1}^{l}(a_i - a_i^*)\phi(x_i) \bullet \phi(x) + b. \tag{16}$$

By introducing kernel function $K(x_i, x_j)$ Eq. (16) can be rewritten as follows:

$$f(x) = \sum_{i=1}^{l}(a_i - a_i^*)K(x_i, x) + b \tag{17}$$

The value of $K(x_i, x_j)$ is equal to the inner product of two vectors. x_i and x_j in the feature space $\phi(x_i)$ and $\phi(x_j)$, that is, $K(x_i, x_j) = \phi(x_i) \bullet \phi(x_j)$. By the use of kernels, all necessary computations can be performed directly in input space, without having to compute the map $\phi(x)$. Some popular kernel functions are shown in Table 1. Using different kernel functions, one can construct different learning machines with arbitrary types of decision surfaces.

Based on the Karush-Kuhn-Tucker (KKT) conditions of quadratic programming, only a certain number of coefficients $(a_i - a_i^*)$ in Eq. (8) will assume non-zero values. The data points associated with them have approximation errors equal to or larger than ε and are referred to as support vectors. These are the data points lying on or outside the ε-bound decision function. According to Eq. (8), it is evident that support vectors and

Table 1 Popular kernel functions

Linear Kernel	$K(x_i, x_j) = x_i \bullet x_j$	
Polynomial Kernel	$K(x_i, x_j) = (x_i \bullet x_j + 1)^d$	d
RBF Kernel	$K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$	$\gamma > 0$
Sigmoid Kernel	$K(x_i, x_j) = \tanh(b(x_i \bullet x_j) + c)$	b, c

thus the sparser the representation of the solution. However, a larger ε can also depreciate the approximation accuracy placed on the training points. In this sense, ε is a trade-off between the sparseness of the representation and closeness to the data (Cao et al., 2003).

APPLYING SVM IN BUS ARRIVAL TIME PREDICTION

SVM, as ANN, does not require a specific form of function. This eliminates the need of function development and parameter estimation for nonlinear and time-varied systems. A well-trained SVM could capture complex relationships between the dependent variables (output such as bus arrival time) and a set of explanatory/independent variables (input such as traffic conditions and passenger demand). Therefore, SVM could be very useful in prediction when it is difficult or even impossible to mathematically formulate the relationship between the input and output such as transit operation.

Using the kernel function the data can be mapped implicitly into a feature space and hence very efficiently. Representing the mapping by simply using a kernel is called the *kernel trick* and the problem is reduced to finding kernels that identify families of regressing formulas. Most of the previous researches selected Gaussian function which is included in radial-basis function (RBF) as the kernel model for regression. The RBF kernel nonlinearly maps samples into a higher dimensional space and, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. Furthermore, the linear kernel is a special case of RBF as Keerthi and Lin (2001) showed: the linear kernel with a penalty parameter C had the same performance as the RBF kernel with some parameters (C, ε, γ). In addition, the sigmoid kernel behaves like RBF for certain parameters (Lin and Lin 2003). Another reason can be the number of hyperparameters, which influences the complexity of model selection. The polynomial kernel has more hyperparameters than the RBF kernel. Hence, the RBF kernel has less numerical difficulties in contrast to polynomial kernels whose values may go to infinity or zero. Moreover, it is noted that the sigmoid kernel is not valid (i.e., not the inner product of two vectors) under some parameters (Vapnik 2000). Therefore, RBF kernel is selected in this study.

The bus travel time will be changed when running on different conditions (time-of-day or weather). According as time-of-day and weather, four patterns are selected as the study patterns for developing prediction model, which correspond to peak time (6:30–7:30) and sunny day (SP), off-peak time (10:00–11:00) and sunny day (SO), peak time and rainy day (RP), off-peak time and rainy day (RO), respectively. Then, the SVM models for four patterns are developed. The proposed SVM models are in the structure shown in Figure 2. The input vector consists of three input variables, segment (x^1), the travel time of current segment (x^2), and the latest travel time of next segment (x^3). The outputs of the models are the predicted travel time between two adjacent time points (y). The arrival time can then be calculated based on departure time and the estimated travel time. The objectives of the prediction models for bus arrival time are to generalize the relationship of the following form:

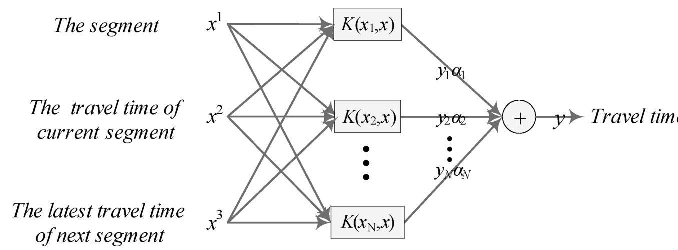

Figure 2 The framework of the proposed SVM models.

$$y = f(x^1, x^2, x^3) \tag{18}$$

The x^1 is self-explanatory. The x^2 denotes the travel time of this running bus on current segment ($k-1 \rightarrow k$) which is used to show the bus currently operational conditions, and x^3 denotes the travel time of the preceding bus on following segment ($k \rightarrow k+1$) which are expected to estimate the traffic conditions of the following segment. This makes SVM, which is based on historical trip data and has the dynamic feature of adjusting prediction, account for the impact of unexpected delays during the trip. For instance, we predict the travel time of bus m on segment $k \rightarrow k+1$. When bus m arrives at stop k, according to $x^1_{k\rightarrow k+1}$, $x^2_{m,k-1\rightarrow k}$ and $x^3_{k\rightarrow k+1}$, the travel time from stop k to stop $k+1 y_{m,k\rightarrow k+1}$ is predicted. Simultaneously, the latest travel time from stop $k-1$ to stop k $x^3_{k-1\rightarrow k}$ is updated, i.e. $x^3_{k-1\rightarrow k} = x^2_{m,k-1\rightarrow k}$, which is used as the input variable of the following buses to predict. As the bus proceeds along its route, the prediction is updated whenever the most recent arrival information is obtained. The process is repeated till the bus reaches the final destination.

NUMERICAL TEST

The model for bus vehicle arrival time prediction has been tested with the data of transit route number 4 in Dalian economic and technological development zone in China. We describe the type of data used in the model first and then the results obtained.

The transit route number 4 goes from the centre to suburb of Dalian development zone, which starts from Xinglinxiaoqu and ends at Lingangxiaoqu with total of 17 bus stops per direction. The total distance of the transit route number 4 extends 13.1 km and the travel time from origin to destination is about 42 minutes. In total, there are 10 time points located along the bus route as shown in Figure 3.

Firstly, it was necessary to divide all the data into different sets according as four patterns (SP, SO, RP, and RO). The collected data consist of the arrival time of the 10 time points in each individual trip at peak time and off-peak time during weekdays. There are 270 valid trips within this one-month period, which consist of 160 trips under SP pattern, 70 trips under SO pattern, 22 trips under RP pattern, and 18 trips under RO pattern. There

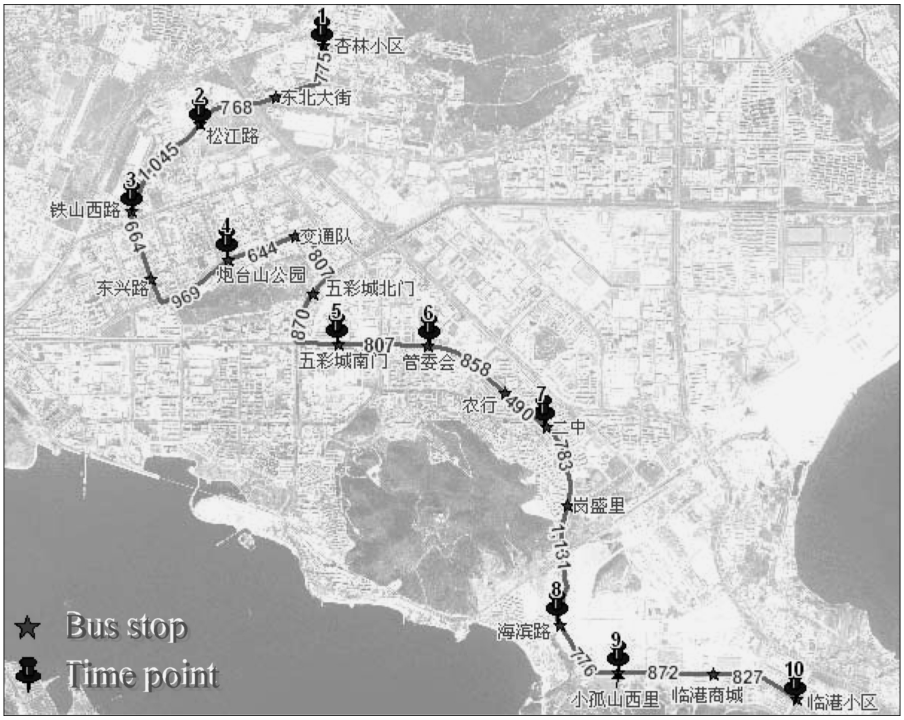

Figure 3 Configuration of transit route number 4.

are 3,480 segment travel time in total. In the data, 250 samples were trained, which consist of 146 trips under SP pattern, 60 trips under SO pattern, 18 trips under RP pattern, and 14 trips under RO pattern. The other samples were set aside as testing data. Then all the training and testing data sets are scaled to [0, 1]. Finally, we developed the four SVM models according as different patterns.

There are three parameters while using RBF kernels: C, ε, and γ. However, it is not known a priori which C, ε, and γ are the best choice for the problem; consequently, some kind of model identification (parameter search) should be investigated. The goal is to identify good parameters (C, ε, γ) to predict unknown data accurately. Normally, to precisely reflect the performance on regressing unknown data and prevent the over-fitting problem the K-fold cross-validation approach is used. The K-fold cross-validation approach splits the available data into more or less K equal parts. $K - 1$ parts of the data will be used to find the SVM estimator, and to calculate the validation error of the fitted model while predicting the kth part of the data. The procedure then continues for $k = 1, 2, \ldots, K$, and the selection of parameters is based on minimum prediction error estimates over all K parts. In this study five-fold cross-validation is conducted as recommended by Hastic and colleagues (2001) with the use of $K = 5$ or 10. In addition to properly selecting the three parameters, there are several methods developed to identify the best C and ε, among which, grid-search is frequently used as the most reliable but a complex one (Hsu et al. 2003; Dong et al. 2005). In 'grid-search', all pairs of (C, ε, γ) are tried and the one with the best performance is picked up. While Hsu (Hsu *et al.* 2003) pointed out that exponentially growing sequences tries of C and ε was a practical way to identify good parameters, $C = 2^{-5}, 2^{-3}, \ldots, 2^{5}$, $\varepsilon = 2^{-13}, \ldots, 2^{-1}$ is considered in this study. For the bus arrival time prediction problem the three parameters are selected as $(2^{-2}, 2^{-5}, 1.58)$.

To validate the input variables of the proposed SVM models, the data of SP pattern are used due to its popularity among operations. The variations between these SVMs and actual travel time are compared. The prediction accuracy was evaluated by computing the root mean squared error (RSME) of each bus route segment. Then, the various SVMs are implemented with SVM MATLAB toolbox (University of Southampton, http://www.isis.ecs.soton.ac.uk /resources/svminfo/, 2005) on Microsoft Windows plat. The results are shown in Figure 4. It can be observed that the performance of the SVM with three input variables is best. Also, the travel time of next segment has more impact than travel time of current segment. In addition, since the SVM-x^1

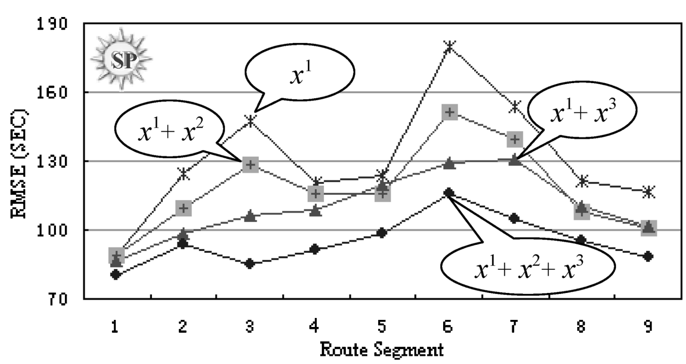

Figure 4 Comparative analysis of in various SVMs.

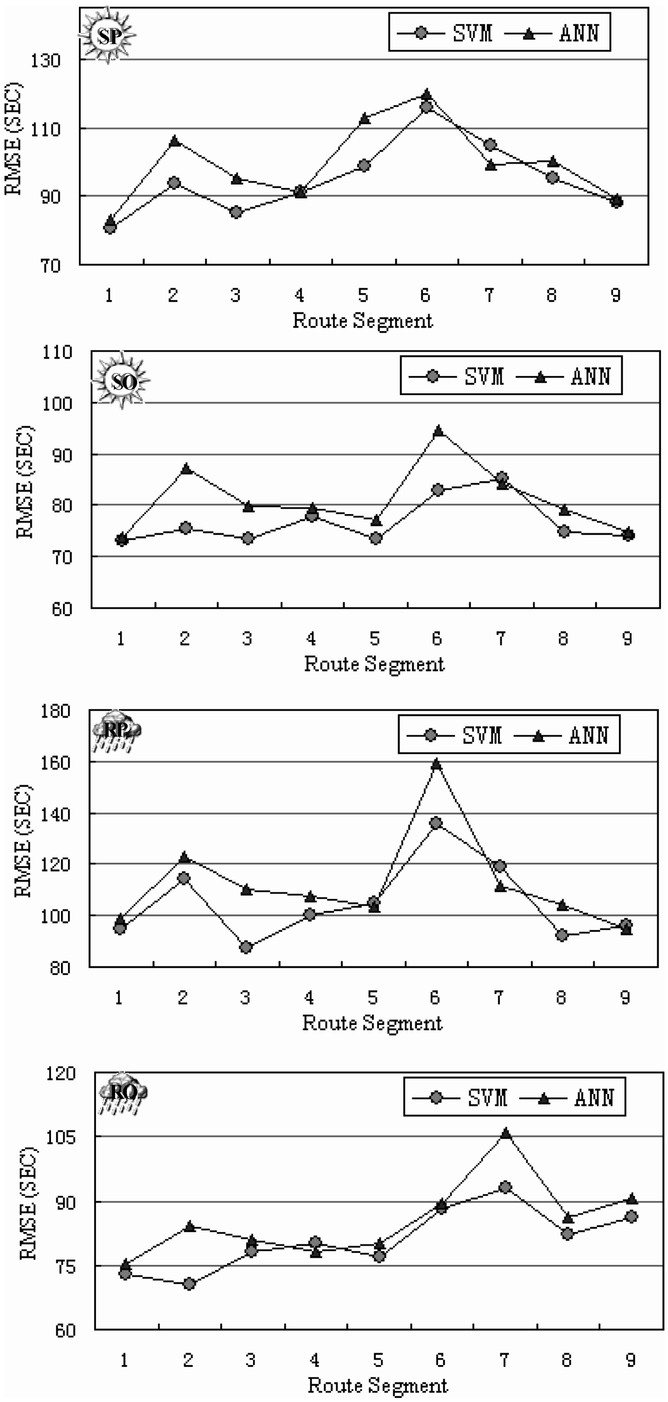

Figure 5 Performance of SVM compared to ANN.

is only based on historical data and has not the real time input variable, its RMSEs shows a more fluctuant trend.

Finally, to evaluate the performance of the proposed model, a standard ANN with three-layer is constructed. The three-layer ANN model is constructed using the same input variables to the proposed SVMs with three input variables to compare the performance of them in this case. A scaled conjugate gradient algorithm (Moller 1993) is employed for training, and the hidden neurons are optimized by trial and error. The final ANN architecture consists of three hidden neurons. In order to have the same basis of comparison, the same training and verification sets are used for both models. Figure 5 shows the comparison of RMSEs of the two models on the four patterns. We can see that the prediction model RMSEs at peak time are much more than these at off-peak time. This can be attributed to the congestion of traffic at peak time, especially in rainy days, which caused transit vehicle arrivals to deviate from the schedule and the vehicle arrival time with reasonable accuracy not to predict. Then, we compare the performance of the SVM with the ANN. It is demonstrated that the SVM exhibits some advantages over ANN model. On the four patterns the SVM outperforms ANN by 5.26%, 5.74%, 7.07%, and 5.91% for the verification data, respectively. Also, the RMSEs of SVMs are generally stable with only few small scale hikes. Thus, compared with the ANN, the SVMs generally provide better indication of the bus arrival time between two adjacent time points. This can be attributed that the SVMs implement the structural risk minimization principle, while ANN models implement the empirical risk minimization principle. The solution of the SVMs may be globally optimal, while ANN models may tend to fall into a local optimal solution. At the same time over-fitting is unlikely to occur with the SVM, if the parameters are properly selected. So the SVM seems to be a powerful alternative for bus arrival time prediction.

Additionally, comparison of SVM to simple regression model was performed. Specifically, we estimated a simple regression model using the following explanatory variables (the travel time of current segment, x^2, and the latest travel time of next segment, x^3). However, the model had weak statistical properties (i.e., statistical significance and fit). Therefore, we have not added the comparison between it with SVM. The regression results are available from the authors, upon request.

CONCLUSIONS

One of the major stochastic characteristics in transit operations is that vehicle arrivals tend to deviate from the posted schedule. Poor schedule or headway adherence is undesirable for both users and operators because it increases passenger waiting/transfer time, discourages passengers from using the transit system, and degrades the operation efficiency and productivity. To predict travel time the consideration of traffic condition is essential. Considering the complexity and difficulties of traffic congestion, this research used the travel time of preceding/current bus on links to estimate traffic conditions of links and developed the prediction models, which consist of the SVM with three input variables, segment, current segment travel time, and latest next segment travel time. The SVM is chosen because of its capability in mapping complicated input/output relations without requiring an explicit function form and good generalizability performance. Tests show that the proposed SVM model in this study can effectively integrate the latest bus information and predict accurate bus arrival time while considering stochastic traffic and demand variation. Furthermore, if real-time data collected from traffic surveillance systems and transit monitoring systems are available, the SVM can be similarly developed to adapt to transit operations in a changeable environment. However, when SVM is applied for solving large-size problems, a large amount of computation time will be involved. In addition, the methods for selecting input variables and identifying the parameters should be further researched.

ACKNOWLEDGEMENTS

This research is financed by the Doctoral Program Foundation of Institution of Higher Education of China (20050151007).

REFERENCES

Cao, L. J., Francis, E. H., and Tay. (2003). Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on Neural Networks*, **14**(6), 1506–1518.

Cao, L. J. and Tay, F. E. H. (2001). Financial forecasting using support vector machines. *Neural Comput. Appl.*, **10**, 184–192.

Chen, M., Liu, X. B., Xia, J. X., and Chien S. I. (2004). A dynamic bus-arrival time prediction model based on APC data. *Computer-Aided Civil and Infrastructure Engineering*, **19**, 364–376.

Chien, I-Jy., Ding, Y., and Wei, C. (2002). Dynamic bus arrival time prediction with artificial neural networks. *Journal of Transportation Engineering, ASCE*, **128**(5), 429–38.

Cristianini, N. and Taylor, J. S. (2000). An introduction to support vector machines and other kernel-based learning methods. New York: Cambridge University Press.

Dailey, D., Maclean, S., Cathey, F., and Wall, Z. (2001). Transit vehicle arrival prediction: Algorithm and large-scale implementation. *Journal of the Transportation Research Board*, **1771**, 46–51.

DeLurgio, S. A. (1998). *Forecasting Principles and Applications*. New York: McGraw-Hill.

Ding, Y. and Chien, S. (2000). The prediction of bus arrival times with link-based artificial neural networks, In *Proceedings of the International Conference on Computational Intelligence & Neurosciences (CI&N)—Intelligent Transportation Systems*, pp. 730–733, Atlantic City, NJ.

Dong, B., Cao, C., and Lee, S. E. (2005). Applying support vector machines to predict building energy consumption in tropical region, *Energy and Buildings*, **37**, 545–553.

Federal Transit Administration. (1998). Advanced Public Transportation Systems: The State of the Art. Update '98. *Rep.* Washington, DC: U.S. Department of Transportation.

Hagan, M. T., Demuth, H. B., and Beale, M. (1996). Neural Network Design. Boston: *PWS*.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). The Elements of Statistical Learning: Data Mining, Inference and Prediction. New York: Springer-Verlag.

Hsu, C. W., Chang, C. C., and Lin, C. J. (2003). A practical guide to support vector classification, technical report, *Department of Computer Science and Information Engineering*, Taipei: National Taiwan University. Available at http://www.csie.ntu.edu.tw/∼cjlin/papers/guide/guide.pdf.

Keerthi, S. S. and Lin, C. J. (2001). Asymptotic behaviors of support vector machines with gaussian kernel, *Neural Computation*, **15**(7), 1667–1689.

Lawrence, S., Giles, C. L., and Tsoi, A. C. (1997). Lessons in neural network training: Overfitting may be harder than expected. In Kuipers, B., Webber, B., and Patil, R. (Eds.), *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, AAAl-97, pp. 540–545. AAAl Press: Mento Park, CA.

Lin, H. T. and Lin, C. J. (2003). A Study on Sigmoid Kernels for SVM and the Training of Non-PSD Kernels by SMO-type Methods, Technical report. Department of Computer Science and Information Engineering, *National Taiwan University*. Available at http://www.csie.ntu.edu.tw/∼cjlin/papers/tanh.pdf.

Moller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, **6**, 523–533.

Moody, J. E. (1992). The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems. *NIPS*, **4**, 847–854.

Mukherjee, S., Osuna, E., and Girosi, F. (1997). Nonlinear Prediction of Chaotic Time Series Using Support Vector Machines. In NNSP'97: Neural Networks for Signal Processing VII: Proceedings of the IEEE Signal Processing Society Workshop. Amelia Island, FL.

Muller, K. R., Smola, J. A., Ratsch, G., Scholkopf, B., and Kohlmorgen, J. (1999). Using support vector machines for time series prediction. In Scholkopf, B., Burges, C.J.C., and Smola (Eds.). Advances in Kernel Methods—Support Vector Learning, Cambridge: MIT Press, MA, pp. 243–254.

Muller, K. R., Smola, J. A., Ratsch, G., Scholkopf, B., Kohlmorgen, J., and Vapnik, V. N. (1997). Predicting time series with support vector machines. In ICANN'97: Proceedings of the Seventh International Conference on Artificial Neural Networks. Lausanne, Switzerland. pp. 999–1004.

Okutani, I. and Stephanedes, Y.J. (1984). Dynamic prediction of traffic volume through kalman filtering theory. *Transp. Res.*, **18B**(1), 1–11.

Park, D. and Rilett, L. R. (1999). Forecasting freeway link travel times with a multilayer feedforward neural network. *Computer-Aided Civil and Infrastructure Engineering*, **14**(5), 357–367.

Ren, J. T., Ou, X. L., Zhang, Y., and Hu, D. C. (2002). Research on networklevel traffic pattern recognition. In *Proc. IEEE 5th Int. Conf. Intelligent Transportation Systems*, 500–504.

Reyna, R., Giralt, A., and Esteve, D. (2001). Head detection inside vehicles with a modified SVM for safer airbags. In *Proc. IEEE 4th Int. Conf. Intelligent Transportation Systems*, 500–504.

Sansom, D. C., Downs, T., and Saha, T. K. (2002). Evaluation of support vector machine based forecasting tool in electricity price forecasting for Australian National Electricity Market participants, *J. Elect. Electron. Eng. Austr.*, **22**, 227–234.

Sarle, W. S. (1995). Stopped training and other remedies for overfitting. In Proceedings of the twenty-seventh symposium on the interface of computing science and statistics, 352–360.

Shalaby, A. and Farhan, A. (2003). Bus travel time prediction model for dynamic operations control and passenger information systems. *The 82nd Annual Meeting of the Transportation Research Board*, Washington, DC. CD-ROM.

Smith, B. L. and Demetsky, M.J. (1995). Short-term traffic flow prediction: Neural network approach, *Transportation Research Record*, **1453**, 98–104.

Stephanedes, Y. J., Kwon, E., and Michalopoulos, P. (1990). Online diversion prediction for dynamic control and vehicle guidance in freeway corridors. *Transportation Research Record*, **1287**, 11–19.

Sun, Z., Bebis, G., and Miller, R. (2002). Improving the performance of on-road vehicle detection by combining Gabor and wavelet features. In *Proc. IEEE 5th Int. Conf. Intelligent Transportation Systems*, 130–135.

Tay, F. E. H. and Cao, L. J. (2001). Improved 3nancial time series forecasting by combining support vector machines with self-organizing feature map. *Intelligent Data Analysis*, **5**, 1–16.

University of Southampton (2005). *Electronics and computer science: Support vector machines*. http://www.isis.ecs.soton.ac.uk/ resources/svminfo/.

Vapnik, V. N. (1999). An overview of statistical learning theory, *IEEE Transactions on Neural Networks*, **10**(5), 988–999.

Vapnik, V. N. (2000). The nature of statistical learning theory. New York: Springer.

Wei, C. and Wu, K. (1997). Developing intelligent freeway ramp metering control systems, *Conf. Proc., National Science Council in Taiwan,* **7C**(3), 371–389.

Weigend, A. (1994). On overfitting and the effective number of hidden units. In *Proceedings of the 1993 Connectionist Models Summer School*, 335–342.

Wu, C. H., Ho, J. M., and Lee, D. T. (2004). Travel-time prediction with support vector regression, *IEEE Transactions On Intelligent Transportation Systems*, **5**(4), 276–281.

Yang, H., Chan, L., and King, I. (2002). Support vector machine regression for volatile stock market prediction, In *Proc. Intelligent Data Engineering and Automated Learning*, ed. H. Yin, N. Allinson, R. Freeman, J. Keane, and S. Hubbard, 391–396. New York: Springer LNCS 2412.