ISSN: 0798-1015

REVISTA ESPACIOS

Vol. 41 (40) 2020 • Art. 9

Recibido/Received: 16/03/2020 • **Aprobado**/Approved: 17/09/2020 • **Publicado**/Published: 22/10/2020

Design of advanced queue system using artificial neural network for waiting time prediction

Diseño de un sistema avanzado para predecir tiempos de espera utilizando una red neuronal artificial

ANUSSORNNITISARN, Pornthep[1]
LIMLAWAN, Varis[2]

Abstract

Queues are common in services such as banking and other services. Unfortunately, it is impossible to eliminate waiting time in a queue since it is very costly to do so. This research proposed an advanced queue system that can provide an accurate waiting time for each customer to increase customer satisfaction. We have applied to the systems an Exponential Weight Average control chart and Artificial Neural Network (ANN). The results show that ANN outperforms other predictors.

key words: waiting time prediction, artificial neural network, queue management system

Resumen

Las colas de espera son habituales en servicios. Desafortunadamente es imposible eliminar las filas de espera debido al alto costo que implica. Esta investigación propone un sistema avanzado de listas de espera que puede proveer el tiempo certero de espera para que esté satisfecho más cada cliente. Hemos aplicado al sistema gráficos de control del Promedio Móvil Ponderado Exponencialmente y Una red neuronal artificial (ANN). Resultados se muestran que la ANN supera el rendimiento de otros predictores.

Palabras clave: Predicción de tiempos de espera, Red Neuronal Artificial, Sistema de gestión de colas.

1. Introduction

1.1. Background

Queue is a part of services and is common in our daily activities such as buying a cup of coffee, waiting for public transports, depositing cash at a bank. Waiting in queue for services is unavoidable for customers, and long waiting in queue has a negative effect on customers' satisfaction (Pruyn & Smidts, 1998; Taylor, 1994). Unfortunately, it is impossible or unwise to eliminate waiting time in queue since it is very costly. For example, a supermarket cannot open more cashier registers to reduce waiting time at peak hours because of area and labor limitations.

Therefore, many businesses allow customers to wait for a service but try to reduce their dissatisfaction by providing certain hospitalities while waiting such as showing a television program, offering coffee, playing music (Kalz et al., 1991). Some business provides a queue-length to customers for increasing their satisfaction and

[1] PhD. Assistant Professor. Industrial Engineering Department, Faculty of Engineering, Kasetsart University Thailand. fengpta@ku.ac.th

[2] PhD. Student. Industrial Engineering Department, Faculty of Engineering, Kasetsart University Thailand. varis.lw@gmail.com

confidences. However, queue-length given to customers is often larger than the actual queue-length because some customers have reneged or left from the queue before receiving service, and the system does not update queue-length. Customers can estimate time to be serviced from the given queue-length, and decide to quit, aka customer bulk. Customer bulk means losing customers to the business.

To provide the estimated waiting time to each arrival customer, ticket queue system has been implemented to manage queue system. In the ticket queue, arrival customers receive the ticket which contain queue information such as queue-length and the estimated waiting time, and then the customers wait for calling at the waiting area. Customers knowing the estimated waiting time can manage their activities while waiting. However, the estimated waiting time given by the ticket queue system is often inaccurate because the queue-length known by the system is often larger than the actual queue-length (Jennings and Pender, 2016; Xu et al, 2007). The ticket queue system does not update queue-length when the customers leave the system. The overestimation of waiting time leads to the customer abandonment problem.

To increase customer satisfaction, it is advisable to provide an accurate time so the customer can spend time doing other activities instead of waiting for the service in the waiting area. This paper aims to propose the design of advanced queue system which can provide the accurate time for service to the customers. Then, customers can manage their time during waiting more efficiently. For example, customers may have lunch at a restaurant instead of being waiting at the shop.

1.2. Review of Waiting Time Prediction

Providing an accurate waiting time to each customer is important in our proposed system, so waiting time predictors are reviewed and discussed as follows:

Queue-Theory based Predictor (QT) was proposed by Whitt (1999). QT is based on n-server queue models where arrival is a general distribution and service times are exponential distribution. QT predicts waiting time of each customer by using queue-length, number of servers and an average service rate. Let $\widehat{W}_{QT}$be the waiting time estimated by QT, *Q(t)* be queue-length at time *t*, *u* be the service rate trained from the historical data (customer/min), s(t) be the number of servers at time *t*. The waiting time predicted by QT is:

$$\widehat{W}_{QT} = \frac{Q(t)+1}{s(t)u}$$

QT is improved by including renege rate in the predictor (Whitt, 1999). The improved QT is called QTR. Let $\widehat{W}_{QTR}$ be the waiting time estimated by QTR, θ be the renege rate, *r* be the number of renege, w_i be the waiting time of customer *i*, *N* be the total customer in training data. Renege rate and QTR predictor are defined as follows:

$$\hat{\theta} = \frac{r}{\sum_{i=1}^{N} w_i}$$

$$\widehat{W}_{QTR} = \sum_{k=0}^{Q(t)+1} \frac{1}{s(t)u + k\theta}$$

QT and QTR are not complexed and are easy to be implemented in waiting time prediction, but the predictors may not perform well in the queue system which has customer abandonment. Queue-length known by the system is often larger than the actual queue-length because queue-length is not updated when customers leave from the system. Incorrect queue-length may affect the accuracy of both predictors. Moreover, the service rate and renege rate may shift from the historical data, so the performance of QT and QTR may decrease because QT and QTR are incapable of changing service rate and renege rate by themselves.

The historical based predictor was proposed in 2008 (Ibrahim & Whitt, 2008, 2009). The waiting time is predicted by using the waiting time of previous customers. There are several historical based predictors, but only one predictor, called last customer to enter service (LES) will be discussed in this paper. The waiting time estimated by LES~~.~~ (W_{LES}) is the waiting time of the latest customer who enters the service. LES can be predicted in any queue model because the predictor does not depend on any queue model and queue-length.

Sometimes the current queue environment such as queue-length is not the same as the past environment. Waiting time estimated by LES may not be accurate for a new customer. For example, the last customer must wait for 10 customers before receiving his/her service, but a new customer may have to wait for only 2 customers in queue. It is common to assure that the waiting time of a new customer is shorter than the previous customer.

Linear Regression (REG) is generally used in many applications (Montgomery & Runger, 2010). In waiting time predictors, the regression spline which is a variant of waiting time has been applied for waiting time prediction in multi-skill call centre (Thiongane et al., 2015). Queue-length and number of servers are used as the inputs of the regression, and the call centre data are generated from 100 simulation replications for training and testing the regression. The regression predictor performs well and better than queue-theory predictors.

Another variant of linear regression for waiting time prediction is quantile regression which uses specific percentiles of responses instead of the mean of responses (Sun et al., 2012). The quantile regression has been used to predict the waiting time in an emergency room department (ED). Approximately 13,200 patients of the Emergency Department in Singapore in January 2011 were used to train and develop the regression model, and the patient data in the Emergency Department from April-June 2011 were used to test the regression model. The inputs of the regression are queue-length, flow rate, and time of day. Only 75% of the predicted waiting time was correct within 22.5 minute. Normally, queue-length has a linear relationship with the waiting time, but flow rate and time of day may not have linear relationship with the waiting time, so the result of the regression is not accurate.

By its nature, linear regression is limited to the linear relationship between a dependent variable and an independent variable, so REG may not perform well if the waiting time and queue-length or other variables have non-linear relationship between them. Moreover, incorrect queue-length and changing in service rate may decrease the accuracy of REG as QT and QTR predictor.

In this paper, we use the simple linear regression (REG) for predicting waiting time. Queue-length divided by the number of servers is predictor variable, and the waiting time is a dependent variable. The waiting time estimated by linear regression is as follows:

$$\widehat{W}_{REG} = \beta_0 + \beta_1 \frac{Q(t)}{s(t)}$$

where β_0 , β_1 are unknown regression coefficients, *Q(t)* is Queue length at time *t* and *s(t)* is the number of server at time *t*

Artificial Neural Network (ANN) is inspired by biological nervous system such as the human brain. ANN simulates how the brain transmit the information through neural network. ANN has been used in many problems such as prediction, clustering and pattern recognition etc. (Abiodun et al., 2018 Pouyanfar et al., 2018).

In waiting time prediction, ANN is applied to predict the waiting time for customer in the call-center service (Thiongane et al., 2015). The architecture of ANN is a simple feedforward where the information flow forwardly from the input node to the output node, and the inputs of ANN are queue-length and the number of servers is the same as in QT and REG predictors. The simulation of the call center is run 100 replication for training and testing. The results of ANN are slightly better than the other predictors such as REG and QT in the queue system with no abandonment, but the accuracy of ANN is not outstanding in the queue system with customer abandonment.

Moreover, the feedforward ANN has also been applied to predict the waiting time of customers at the bank (Satya et al., 2018). Apart from queue-length and the number of servers, the waiting time of previous customers is used as the input of ANN to improve the accuracy of prediction. Approximately 22,082 data points from a bank in Indonesia have been used for training and testing ANN. The result shows that ANN with the previous waiting time is slightly better than ANN with queue-length and the number of servers.

In those works, ANN can predict an accurate waiting time for each customer, and the inputs of ANN can be adjusted for improving the accuracy of waiting time prediction. However, there is no work considering customer abandonment which may affect the accuracy of waiting time prediction. In the queue system with customer abandonment, customers may leave the queue before receiving the service, and the system in those studies does not update queue-length. The incorrect queue-length affects the accuracy of predicted waiting time of ANN.

Unlike those works, queue-length in our proposed system is constantly updated when the customers exit the system. Moreover, the service rate of the system is also monitored and updated constantly instead of using the service rate from the historical data. ANN with our system is expected to have more accuracy.

2. Methodology

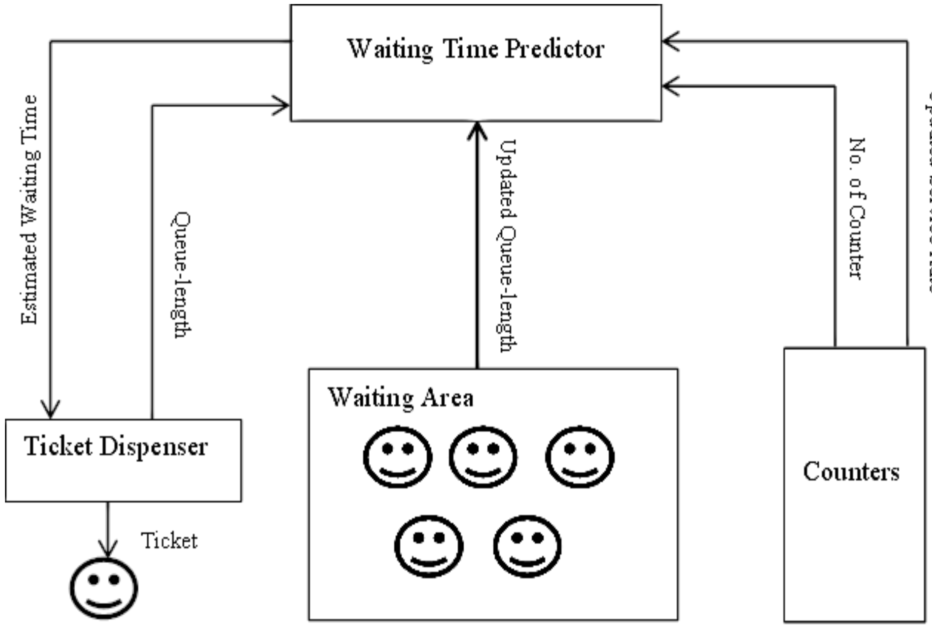
2.1. The Proposed Queue Management System

Based on the disadvantage of the ticket queue technology, an advanced queue system that can provide an accurate estimated waiting time is proposed. The design of the proposed system was developed as shown in figure 1. The proposed system constantly monitors and updates queue-length and service rate. The updated queue-length and service rate are sent to the waiting time predictors to estimate the waiting time to each customer. The detail of the system is described as follows:

Nowadays, video surveillances are easy to be installed because the cost has decreased; therefore, we can find video surveillance in many places such as banks, houses and streets. For advanced purpose, video surveillance is installed with computer vision and has been applied in many applications such as car detection and people counting. Computer vision is an advanced image processing computerized system that can automatically perform tasks from images or videos with the help of machine learning and artificial intelligence.

The technologies are applied to predict the waiting time in queue. For example, Video Surveillance and Computer Vision were applied to collect the number of people, the number of opened-counter (Culjal et al., 2012; Parameswaran et al., 2012). Wi-Fi and smartphone were used to collect queue information such as queue-length and arrival rate (Shu et al, 2016; Uhler et al., 2012). The availability of technologies allows us to design the system which can easily track queue-length. Queue-length collected from the technologies is expected to have more accuracy, but the detail of the technogies will not be discussed in the detail. In this paper, We assume that queue-length collected from the technologies is equal to the actual queue-length in the experiments.

Figure 1
Diagram of the Proposed Advanced Queue System



In the proposed system, an average service time is also monitored and detected. If the average service time is shifted from the historical data, the system will detect and adjust the average service rate. Exponential Weight Average (EWMA) control chart which is a tool for monitoring and detecting mean shift is applied to the system (Montgomery, 2012).

Assume that u_0 be an initial service rate from the historical data, σ be the standard deviation of service rate from the historical data, z_i be service rate of customer *i*, λ and L be EWMA parameters. The implement of EWMA control chart in our system will be described as follows:

Step 1: Set up an initial mean of the service time (u_0), the standard deviation of service time σ, EMWA parameters which are λ and L, an initial predicted service time, $z_0 = u_0$

Step 2: Predict the service time of customer i by using the following formula:

$$z_i = \lambda x_i + (1-\lambda)z_{i-1}$$

Step 3: Calculate the lower bound (LCL) and the upper bound (UCL) as follows.

$$UCL = u_0 + L\sigma\sqrt{\frac{\lambda}{(2-\lambda)}[1-(1-\lambda)^{2i}]}$$

$$LCL = u_0 - L\sigma\sqrt{\frac{\lambda}{(2-\lambda)}[1-(1-\lambda)^{2i}]}$$

Step 4: If z_i is more than UCL or less than LCL, then u_0 equals z_i

Step 5: Repeat step 2 until the system terminates.

Once the system updates queue-length and service time, the updated queue-length and updated average service time will be used as the input of the waiting time predictor. Based on the advantage of ANN, ANN is selected to predict the waiting time in the system. The architecture of ANN implemented in our system is a single hidden layer feedforward with a linear output layer.

Based on Hegan et al. (2014), the single hidden layer feedforward with M hidden neurons has 3 layers which are input layer, hidden layer and the output layer as shown in figure 2. Queue-length, the number of server and service rate are 3 neurons in the input layer, and the predicted waiting time is only neuron in the output layer. The single hidden layer feedforward is mathematically expressed as follows:

$$\boldsymbol{p} = \begin{bmatrix} u(t) \\ Q(t) \\ S(t) \end{bmatrix} \qquad , \text{for } t = 1, 2, 3, \ldots$$

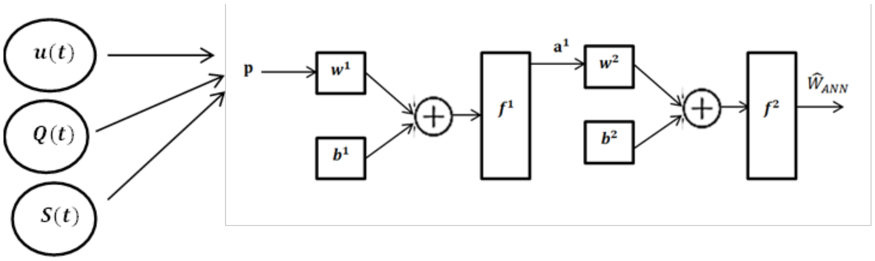
$$\boldsymbol{a}^0 = \mathbf{p}$$

$$\boldsymbol{a}^{i+1} = f^{i+1}(\mathbf{w}^{i+1}\boldsymbol{a}^i + \mathbf{b}^{i+1}) \qquad , \text{for } i = 1$$

$$\mathbf{a}^2 = [\widehat{W}_{ANN}]$$

where *u(t)* is the updated service rate at time *t*, *Q(t)* is the updated queue-length at time t and *S(t)* is the number of server at time *t*, *t* is discrete time, $\mathbf{p}$ is the inputs from the external source, $\boldsymbol{a}^i$ is the input vector of layer *i*, $\mathbf{b}^i$ is the biased vector of layer *i*, $\mathbf{w}^i$ is the weights vector of layer *i*, f^i is the activate function of layer *i*.

Figure 2
A Single Hidden Layer Feedforward ANN



As illustrated in Figure 2, the input **p** is updated by monitoring the system and is sent to the input node of ANN predictor. The inputs flow forwardly from the input node to the output node. The output of output node is the estimated waiting time. The output of layer *i* is the input of layer *i*+1. The output of the last layer is the predicted waiting time $\widehat{W}_{ANN}$.

ANN based predictor is trained by the standard backpropagation algorithm or gradient decent backpropagation which is the supervised learning. In this paper, the algorithm is batch training which uses n historical data in the training set. The algorithm aims to minimize the mean sum of square error in the predictions by adjusting weight and biased. The mean sum of square error is calculated as follows.

$$F(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^{n}(\mathbf{t}_j - \mathbf{a}_j)^T(\mathbf{t}_j - \mathbf{a}_j) = E[\mathbf{e}^T\mathbf{e}]$$

where $\boldsymbol{x}$ is the vector of the weight and biased, $\mathbf{t}_j$ is the vector of actual waiting time of data *j* in the training set, $\mathbf{a}_j$ is the vector of estimated waiting time by ANN of data *j* in training set. $\mathbf{e}$ is the vector of error at iteration *k*, *n* is the total data of the training set.

For each k iterations, the first step is to estimate the output for each layer of ANN. The next step is to calculated sensitivities of each data j backward from the last layer *M* to the first layer. The sensitivities of each layer are expressed as follow.

$$\boldsymbol{s}_j^M = -2\mathbf{F}^M(\mathbf{n}^M)\left(\mathbf{t}_j - \mathbf{a}_j\right)$$

$$\boldsymbol{s}_j^i = \mathbf{F}^i(\mathbf{n}^i)(\mathbf{w}^{i+1})^T\mathbf{s}_j^{i+1}$$

where $\boldsymbol{s}_j^M$ is the sensitivities vector for the last layer *M* of data *j* in training set, $\boldsymbol{s}_j^i$ is the sensitivities vector for layer *i* of data *j* in training set , $\mathbf{F}^M(\mathbf{n}^M)$ is the derivative of the activated function of layer *M*. $\mathbf{F}^i(\mathbf{n}^i)$ is the derivatives of the activated function of layer *i*.

The next step is to update the weights and biased of each layer i by using the following equation.

$$\mathbf{w}^i(k+1) = \mathbf{w}^i(k) - \frac{\mu}{n}\sum_{j=1}^{n} \boldsymbol{s}_j^i\left(\mathbf{a}_j^{i-1}\right)^T$$

$$\mathbf{b}^i(k+1) = \mathbf{b}^i(k) - \mu\sum_{j=1}^{n} \boldsymbol{s}_j^i$$

where $\mathbf{w}^i(k)$ and $\mathbf{b}^i(k)$ are weight and biased of layer *i* in iteration *k* and µ is learning rate. $\mathbf{a}_j^i$ is the output vector of *i* layer of data *j* in training set. The algorithm is run until the mean sum of square error reaches the acceptable level.

To compare ANN with other predictors, QTR and REG are also updated by the proposed system. The updated queue-length and the updated service rate are used as the inputs in QTR, and the updated queue-length is the only input that can be updated in REG. The summary of each predictor is shown in table 1.

Table 1
Summary of the Queue Information used in Each Predictor

Predictor	Queue Information
Queue Theory based Predictor (QT)	1) Queue-length (No update for customer bulk and customer renege) 2) Service rate trained from historical data 3) Number of server
Linear Regression (REG)	1) Queue-length is updated by the proposed system 2) Number of server
Queue Theory based Predictor with Renege Customer (QTR)	1) Queue-length is updated by the proposed system 2) Service rate is updated by the proposed system 3) Number of server
Artificial Neural Network (ANN)	1) Queue-length is updated by the proposed system 2) Service rate is updated by the proposed system 3) Number of server

2.2. Experiment

In this section, the performance of each predictor is evaluated. The experiments setting and result are described as follow.

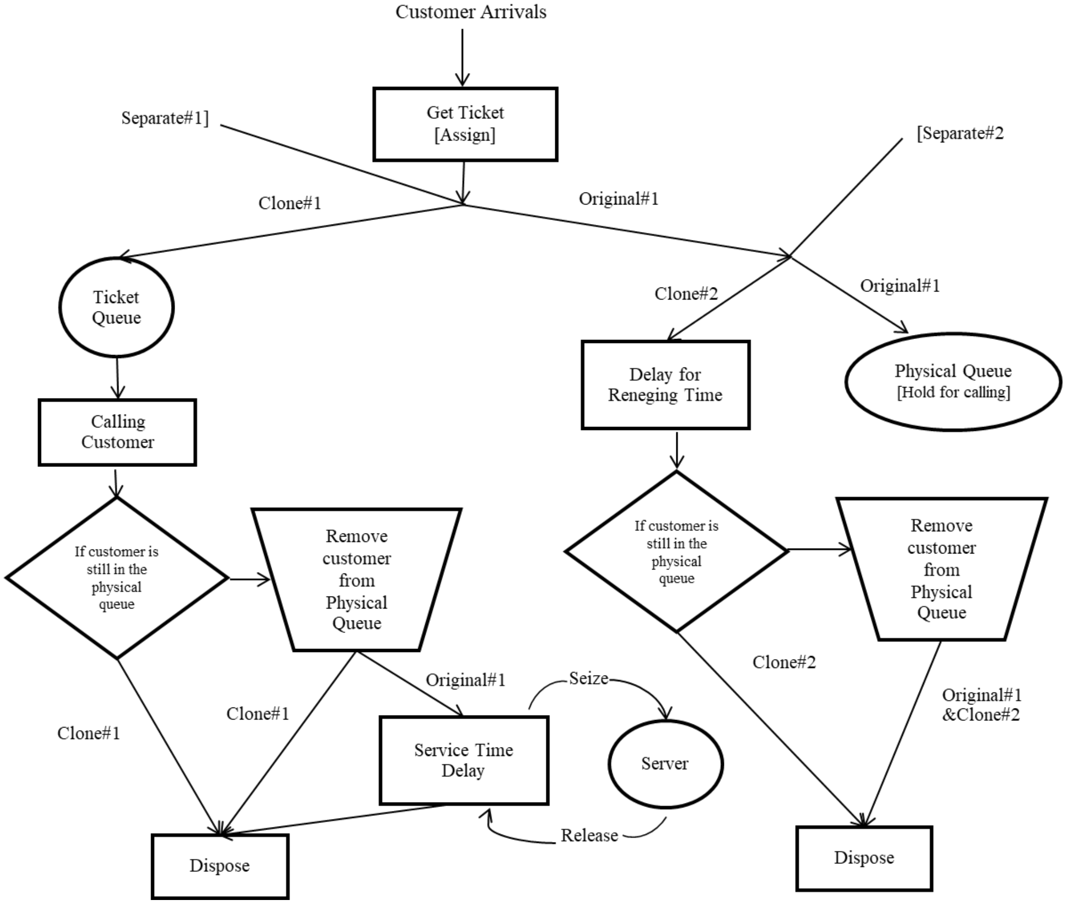
In order to evaluate our system, the ticket queue system is simulated as illustrated in figure 3. Customers arrives and get the ticket and then wait for a calling at the waiting area. The original#1 represents customers' activities at the waiting area. Some customers may leave from the queue. Customer bulk in this model is the customers whose renege time is zero. The clone#2 represented the renege decision for each customer, and the clone#1 represents the flow of queue information. When customers get the ticket, the system updates the ticket queue.

The system will call customers in the physical queue when their queue is reached. In this model, the ticket queue and physical queue are separated and are easily collected.

All test data are dynamic multi-server queue systems with reneging, and are simulated by Version 14.0 of SIMAN ARENA® Rockwell software (2012); . In order to simulate the real-world problem, the simulation systems which represent service organization such as bank and hospital are described as follow.

- The system opens at 8.00 a.m. and closes at 16.00 p.m.
- The arrival rate, the service time, the number of server and the load of the system are dynamically changed in a replication.
- The service rate is a normal distribution with a mean of μ and a standard deviation of 0.75.
- The system load (ρ) reaches its peak during 11.00 AM -1.00 PM.
- The arrival rate is Poisson distribution with a mean of λ which is adjusted based on the load ρ and the mean service rate μ.
- The reneging time of customers is 2-Erlang distribution with a mean of 15.
- The schedule of arrival rate, the service time and the load of the system is shown in table 2.

Figure 3
The Simulation Diagram of the Ticket Queue System Simulation



As shown in table 3, test cases are divided into 7 cases by the size of service time shift. The shift sizes are adjusted from -0.5 to +1.0. Queue parameters such as arrival rate, service time, load of the system are adjusted according to table 2. Each case runs 10 replications for generating test data. The run of training set is 30 replications, and the run of validation and the run of test set are 10 replication. The training data set is created by randomly mixing from case 1 to case 7.

Table 2
The Schedule of the Test Queuing System

Time	Arrival Rate (customer/hour)	Service Time (minute/customer)	No. of Server	Load (p)
8AM – 9AM	POISSION(λ) λ depends on load (p) and avg. service time (μ)	Norm(1, 0.75)	1	0.85
9AM – 10AM		Norm(1, 0.75)	1	0.85
10AM – 11AM		Norm(1, 0.75)	2	0.95
11AM – 12PM		Norm (1 ±shift size, 0.75)	2	0.99
12PM – 1PM		Norm (1 ±shift size, 0.75)	3	0.99
1PM – 2PM		Norm (1 ±shift size, 0.75)	2	0.95
2PM – 3PM		Norm (1 ±shift size, 0.75)	1	0.85
3PM – 4PM		Norm (1 ±shift size, 0.75)	1	0.85

The proposed system and each predictor except ANN are coded by Excel VBA in Version 2016 of Microsoft Excel (2016) which easily manages the dataset. ANN is designed and created by Version R2016a of MATLAB (2015). The neuron in the hidden layer is 5. The activated function of the hidden layer and the activated function of the output is tan-sigmoid function and pure linear function respectively. ANN is trained by backpropagation algorithm (traingd) in Version R2016a of MATLAB (2015). The learning rate (μ) of the backpropagation algorithm is 0.05. λ and L of EWMA control chart in the monitoring system are 0.1 and 2.7 respectively.

In the numerical experiments, we use the mean square error (MSE) to evaluate the performance of each predictor. The error and MSE are as follow.

$e_i = W_i - \widehat{W}_i$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} {e_i}^2$$

where e_i is the prediction error, W_i is the actual delay of customer *i* , $\widehat{W}_i$ is the waiting time predictor of customer *i* and n is the number of the test sample.

Percentage of accuracy within an acceptable level is used to measure the performance of the best predictor. For better measuring the accuracy of the prediction, we set several of acceptance tolerance which is denoted by *l*. The predicted waiting times within the tolerance are counted as the accurate prediction. For example, the tolerance is 5 minutes. If a predicted value differs from the actual value within ±5 minutes, the predicted value is counted as the accurate prediction. Let N be number of test samples, e_i be an error of *i* sample, c_i is the accuracy indicator of *i* sample. The percentage of accuracy is defined as follow.

Percentage of Accuracy within $\pm l\ = \frac{1}{N}\sum_{i=1}^{n} c_i \times 100$

where, $c_i = \begin{cases} 1 & , e_i \in [-l, l\,] \\ 0 & , e_i \notin [-l, l\,] \end{cases}$

Table 3
The Detail of Test Cases

Test Case	Service Time Shift (min)	Queue parameters	#Simulation Run (Replication)	Predictors
1	-0.5	u, λ, ρ are set according to table 1	Training set = 30 Validation set = 10 Test set = 10	1) QT 2) LES 3) REG 4) QTR 5) ANN
2	-0.25			
3	0			
4	+0.25			
5	+0.5			
6	+0.75			
7	+1			

3. Results

3.1. Result of the experiment

As shown in table 4 and figure 4, LES and REG perform worse than QT in 5 out of 7 cases. The percentage of difference from QT of LES is approximately -100% in case 3, case 4, case 5 On the other hand, QT and QTR is better than QT 5 out of 7 cases. The MSE of QT decrease as the shift size increases, especially in case 1. As shown in figure 5, the predicted waiting time of QT and the predicted waiting time of ANN are compared to the actual waiting time in case 1 in which MSE of QT is obviously worse than the others. The waiting time predictions from QT always overestimate the actual value when the service rate has been shifted negatively in case 1. On the other hand, the waiting time estimated from ANN can adapt itself and follow the actual value.

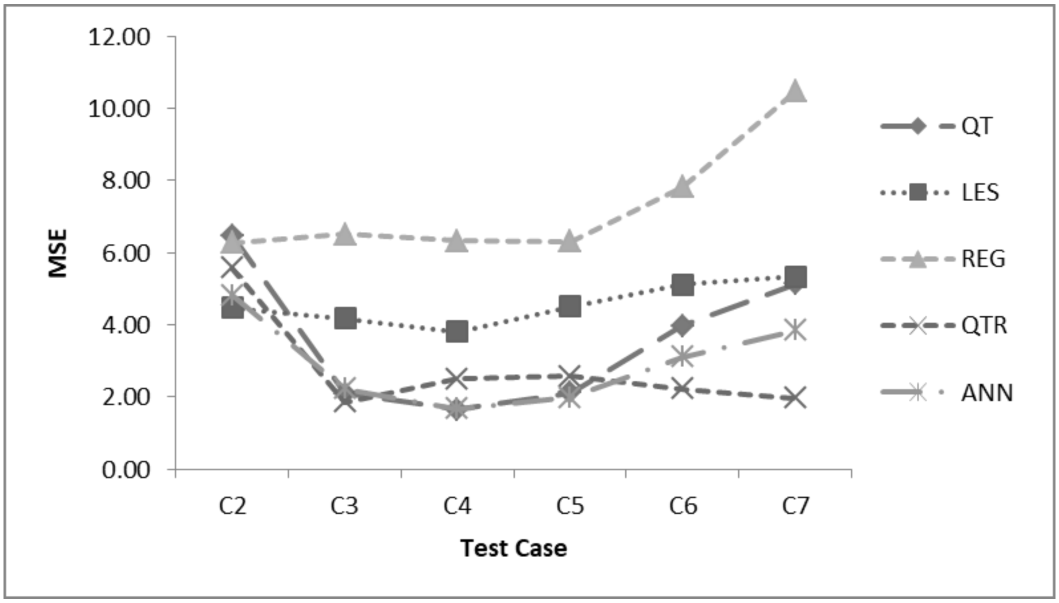
Table 5 summaries the percentage of accuracy within the acceptable tolerance for ANN based predictor which is the best predictor in the experiment. The percentages of accuracy within 1 minute tolerance are more than 50% in case 3 to case 7, but are 41.3% in case 2 and 19.2% in case 1. More than 90% of predicted waiting time is correct within 3 minutes tolerance except case 1 and case 2. The percentages of accuracy within 5 minutes tolerance are more than 95% except case 1, and the percentage of accuracy within 10 minutes tolerance are more than 99% in all cases.

Table 4
Comparison of MSE

Test Case (shift)	QT	LES		REG		QTR		ANN	
	MSE	MSE	%Diff*	MSE	%Diff*	MSE	%Diff*	MSE	%Diff*
1	114.56	8.96	92.2	9.02	92.1	16.01	86.0	13.53	88.2
2	6.48	4.47	31.1	6.28	3.1	5.57	14.0	4.82	25.6
3	2.10	4.17	-98.9	6.52	-210.7	1.87	11.0	2.23	-6.3
4	1.66	3.81	-130.4	6.34	-282.9	2.50	-51.2	1.68	-1.4
5	2.11	4.50	-113.4	6.31	-199.2	2.58	-22.2	1.99	5.7
6	3.97	5.11	-28.5	7.82	-96.7	2.23	43.8	3.12	21.6
7	5.13	5.34	-4.0	10.46	-103.9	1.96	61.8	3.84	25.1

*%Diff is the difference from QT = (MSE of QT – MSE of predictor)/100

Figure 4
Comparison of MSE of Each Predictor in Case 2 to Case 7



3.2. Discussion

On the basis of the result, the proposed system improves the accuracy of the waiting time predictor mentioned in section 1. The updated service rate and updated queue-length has an effect on the performance of the waiting time predictor as we have expected. ANN and QTR are more suitable for the proposed system than REG. For future development, ANN is more appropriate than QTR because the structure of ANN can be adapted for advanced system. For example, the other queue information such as arrival rate, load of the system can be added to ANN's inputs.

The performance of the ANN-based predictor in the positive shift cases is better than the performance of the ANN-based predictor in negative shift cases. The performance in the negative shift case should be improved in the further development. However, ANN is accurate with 5 minutes tolerance, thus we can guarantee the customer that the predicted waiting time will be accurate within 5 minutes. Customers, who know the exact period of waiting time can plan their activities during the waiting or decide to stay or do something else outside the shop.

Figure 5
Plot for Actual Waiting Time and Estimated
Waiting Time by QT and ANN in Case 1

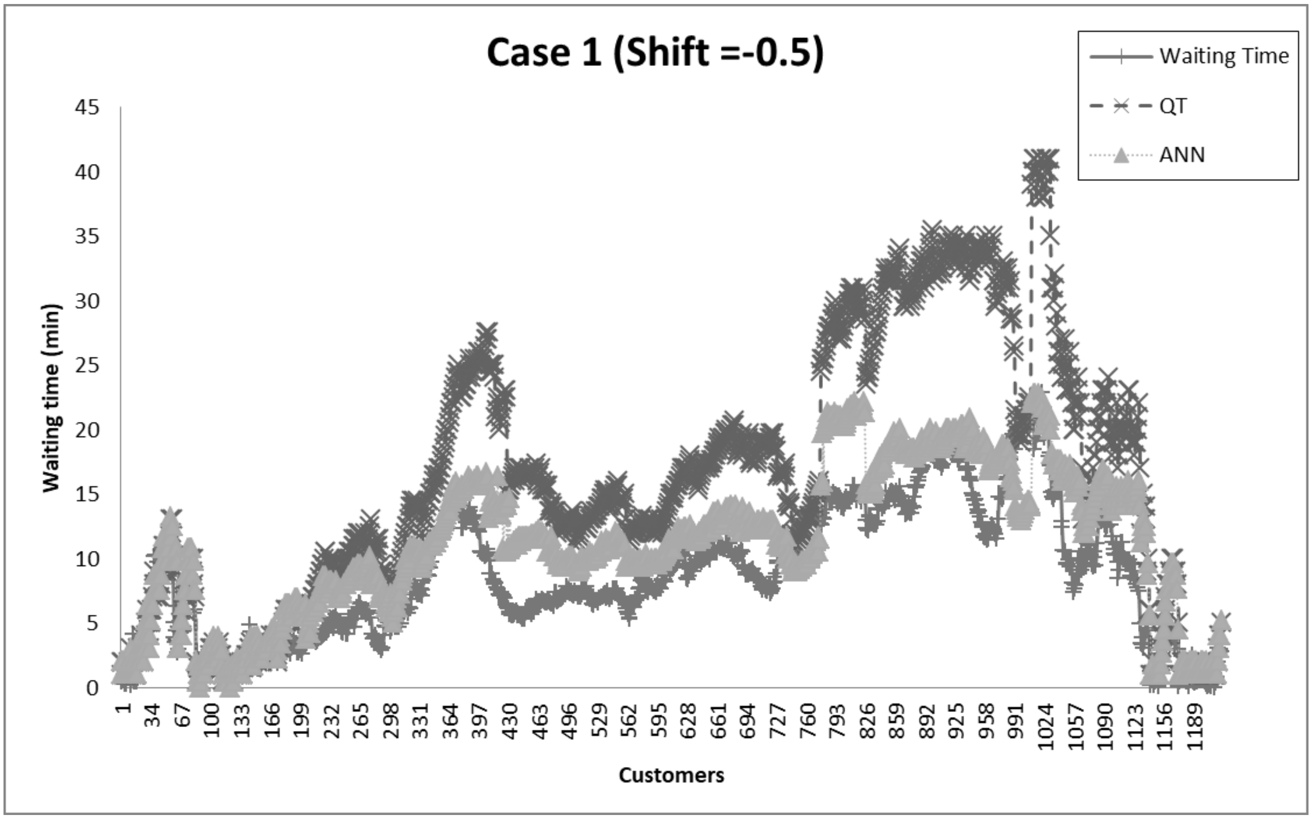


Table 4
Percentage of Accuracy within the Acceptable Tolerance for ANN

Case	Accuracy (%) within ±1 min	Accuracy (%) within ±3 min	Accuracy (%) within ±5 min	Accuracy (%) within ±10 min
1	19.20%	53.60%	83.40%	99.70%
2	41.30%	83.30%	96.70%	99.90%
3	62.00%	94.90%	98.80%	99.90%
4	65.00%	96.50%	99.70%	100.00%
5	61.10%	95.00%	99.50%	100.00%
6	63.80%	92.60%	97.20%	99.80%
7	57.50%	90.80%	96.00%	99.90%
Average	52.84%	86.67%	95.90%	99.89%

4. Conclusions

In this paper, we propose the an advanced queue prediction system which can provide more accurate time to be served to each customer. To improve the accuracy of the waiting time prediction, the proposed system constantly monitors and detects the change in queue-length and service rate. Exponential Weight Average (EWMA) control chart is applied to detect and adjust the change in the service rate. The updated queue-length and updated service rate from the proposed system are used as the input of Artificial Neural Network (ANN) for predicting the waiting time.

ANN with the system is compared to queue-theory-based predictor without the proposed system, queue-theory-based predictor with the proposed system, linear regression and the historical based predictor. ANN with the proposed system outperforms queue-theory-based without the proposed system in most cases. Moreover, ANN and queue-theory-based predictor with updated information performs better than linear regression and historical-based predictor, but the performance of ANN and queue-theory-based predictor with update queue information are not different. The results show that the proposed system can improve the accuracy of waiting time prediction.

To implement the proposed system in practice, camera and computer vision might be applied to collect queue information in the system

This work revisits a classical problem which is the waiting time prediction in queuing systems. The proposed system with artificial neural network and computer vision system can improve the accuracy of waiting time prediction. With the accurate predicted waiting time, customers can manage their activities during waiting for services.

Bibliographic references

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), e00938. doi:https://doi.org/10.1016/j.heliyon.2018.e00938

ARENA Simulation Software [Computer Software] (2012) Retrieved from https://www.arenasimulation.com/

Culjak, I., Abram, D., Pribanic, T., Dzapo, H., & Cifrek, M. (2012, May). *A brief introduction to OpenCV*. Paper presented at the 2012 Proceedings of the 35th International Convention MIPRO, Opatija, Croatia.

Hegan, M.T.,Demuth, H.B., Beale, M.H. and Jess. O.D. (2014). Neural Network Design. Retrieved from https://hagan.okstate.edu/NNDesign.pdf/

Ibrahim, R., & Whitt, W. (2008). Real-time delay estimation in call centers. *Proceedings - Winter Simulation Conference*(pp. 2876-2883). Florida, USA.

Ibrahim, R., & Whitt, W. (2009). Real-Time Delay Estimation Based on Delay History. *Manufacturing & Service Operations Management*, 11(3), 397-415. doi:10.1287/msom.1080.0223

Jennings, O. B., & Pender, J. (2016). Comparisons of ticket and standard queues. *Queueing Systems*, 84(1), 145-202. doi:10.1007/s11134-016-9493-y

Katz, K. L., Larson, B. M., & Larson, R. C. (1991). Prescription for the Waiting-In-Line Blues: Entertain, Enlighten, and Engage. *Sloan Management Review*, 32(2), 44-53.

MATLAB [Computer Software] (2015). Retrieved from https://www.mathworks.com/products/matlab.html

Microsoft Excel [Computer Software] (2016). Retrieved from https://office.microsoft.com/excel

Montgomery, D. C. & Runger, G. C. (2010). *Applied statistics and probability for engineers* (5th ed.). Hoboken, NJ: John Wiley & Sons, Inc.

Montgomery, D.C. (2012). *Introduction to Statistical Quality Control* (7th ed.). Hoboken, NJ: John Wiley & Sons, Inc.

Parameswaran, V., Shet, V., & Ramesh, V. (2012). Design and Validation of a System for People Queue Statistics Estimation. In C. Shan, F. Porikli, T. Xiang, & S. Gong (Eds.), *Video Analytics for Business Intelligence* (pp. 355-373). Berlin, Heidelberg: Springer Berlin Heidelberg.

Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C. and Iyengar, S. S. (2018). A Survey on Deep Learning: Algorithms, Techniques, and Applications. *ACM Comput. Surv*., 51(5), 1-36. doi:10.1145/3234150

Pruyn, A., & Smidts, A. (1998). Effects of waiting on the satisfaction with the service: Beyond objective time measures. *International Journal of Research in Marketing*, 15(4), 321-334. doi:https://doi.org/10.1016/S0167-8116(98)00008-1

Satya Hermanto, R. P., Suharjito, Diana, & Nugroho, A. (2018). Waiting-Time Estimation in Bank Customer Queues using RPROP Neural Networks. *Procedia Computer Science*, 135, 35-42. doi:https://doi.org/10.1016/j.procs.2018.08.147

Shu, H., Song, C., Pei, T., Xu, L., Ou, Y., Zhang, L., & Li, T. (2016). Queuing Time Prediction Using WiFi Positioning Data in an Indoor Scenario. *Sensors (Basel, Switzerland)*, 16(11), 1958. doi:10.3390/s16111958

Sun, Y., Teow, K. L., Heng, B. H., Ooi, C. K., & Tay, S. Y. (2012). Real-Time Prediction of Waiting Time in the Emergency Department, Using Quantile Regression. *Annals of Emergency Medicine*, 60(3), 299-308. doi:http://dx.doi.org/10.1016/j.annemergmed.2012.03.011

Taylor, S. (1994). Waiting for Service: The Relationship between Delays and Evaluations of Service. *Journal of Marketing*, 58(2), 56-69. doi:10.2307/1252269

Thiongane, M., Chan, W., & Ecuyer, P. L. (2015, December). *Waiting time predictors for multi-skill call centers*. Paper presented at the 2015 Winter Simulation Conference, Huntington Beach, CA

Bulut M.F., Yilmaz Y.S., Demirbas M., Ferhatosmanoglu N., Ferhatosmanoglu H. (2013). LineKing: Crowdsourced Line Wait-Time Estimation Using Smartphones. In: Uhler D., Mehta K., Wong J.L. (eds), *Mobile Computing, Applications, and Services. MobiCASE 2012. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol 110. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-36632-1_12

Whitt, W. (1999). Predicting queueing delays*. Management Sci*., 45, 880-888. doi:10.1287/mnsc.45.6.870

Xu, S. H., Gao, L., & Ou, J. (2007). Service Performance Analysis and Improvement for a Ticket Queue with Balking Customers. *Management Science*, 53(6), 971-990. doi:10.1287/mnsc.1060.0660