

Intelligent Bus Application for Smart City based on LoRa Technology and RBF Neural Network

LINDA KANAAN¹, JAMAL HAYDAR², MOUNIR SAMAHA³, ALI MOKDAD⁴, WALID FAHS⁵

CCE Department, Faculty of Engineering
Islamic University of Lebanon (IUL)
Wardanieh, Mount Lebanon
LEBANON

Abstract: Nowadays smart transportation is one of the most important smart city applications for all the facilities it provides, as well, achieves green transportation. One of these applications include bus monitoring systems for schools and universities. This paper proposes an integrated bus tracking and monitoring system using Internet of Things (IoT) mainly LoRa technology for Islamic University of Lebanon (IUL) to ease the bus issue that comprises a barrier for good service. In the proposed system, we have implemented a case study to monitor the position, speed, humidity and temperature of IUL shuttle buses. In addition to the tracking, a prediction algorithm is implemented for computation of the arrival time of the bus to the Wardanieh campus. The estimation is based on artificial intelligence neural network mainly the Radial Basis Function (RBF) algorithms for the sake of finding the predicted time of arrival in different locations and scenarios starting from Khaldeh campus to the Wardanieh campus. The collected data are instantly shown on a user interface for monitoring, whereas the exact location and arrival time of the bus will be displayed on an android application for students. Results show that the arrival estimated time error is about 0.18%.

Key-Words: -Internet of Things, LoRa, artificial intelligence, neural networks, Radial Basis Function, smart city, smart transportation, green transportation, university service.

Received: May 10, 2020. Revised: November 19, 2020. Accepted: November 30, 2020. Published: December 11, 2020.

1 Introduction

The twenty-first century is full of new challenges and capabilities defined by the progress in internet technologies, data science and artificial intelligence. In smart cities, wide-area wireless networks are typically utilized to enable connectivity between a large network of diverse data-generating internet of thing (IoT) devices, sensors, and smart meters. Thus, data-driven decision-making, enhancing city resource management, and public services are enabled due to the insights generated from the information [1]. IoT technologies have a great positive impact on many different areas of urban life including its quality, cost-efficiency, and environmental sustainability. Before IoT, applications were implemented using GSM networks and radio devices such as ultra high frequency devices. Consequently, various smart city applications have been achieved and built using different IoT technologies including NB-IoT, sigfox, LoRaWAN, etc. Table 1 depicts the comparison between different IoT technologies in terms of modulation, frequency, bandwidth, data rate, maximum range and many other parameters. Recently smart transportation is one of the most important smart city applications for all the facilities and safety it provides. In fact, vehicle tracking and monitoring systems have been evolving lately as they provide a decisive help in finding the location or keeping track of vehicles and

drivers. Actually, schools and universities nowadays outside Lebanon agrees on the importance of their student's safety, pleasure and time. In almost all universities in Lebanon, most of the time students and staff are not aware of exact timing and location of the college bus which causes lots of discomfort. This problem is so serious when it comes to critical timings, actually students waste most of their times waiting for the bus to arrive, not having any idea about its arrival time. Moreover, monitoring the bus state and the conditions the students have to deal with are important too. Through the years, we noticed that Islamic University of Lebanon (IUL) students suffer most of the time because of the bus timings, actually they frequently complained about late buses. Moreover, students demanded to know the exact arrival time of the bus so as not to waste time in waiting. This problem really motivated us to find a solution where we can help students stress less about the transportation to and from the university. For the above mentioned reasons and since LoRa technology supports mobility [2] and it is ideal for green smart cities due to its low power consumption, we proposed an integrated bus tracking and monitoring system using LoRa technology for IUL embedded with a prediction system based on neural network mainly radial basis function (RBF) for the arrival time estimation. This project is mainly implemented for IUL university to ease the

Table 1: Comparison between different IoT technologies

Technology	Sigfox	LoRaWAN	NB-IoT
Modulation	BPSK	CSS	QPSK
Frequency	Unlicensed ISM bands (868MHz in Europe, 915MHz in North America, and 433MHz in Asia)	Unlicensed ISM bands (868MHz in Europe, 915MHz in North America, and 433MHz in Asia)	Licensed LTE frequency bands
Bandwidth	100 Hz	250 kHz and 150 kHz	200 kHz
Maximum data rate	100 bps	50 kbps	200 kbps
Maximum payload length	12 bytes (UpLink), 8 bytes (DownLink)	243 bytes	1600 bytes
Range	10 km (urban), 40 km (rural)	5 km (urban), 20 km (rural)	1 km (urban), 10 km (rural)
Adaptive data rate	No	Yes	No

bus issue and facilitate the students' transportation experience. In addition, the proposed system helps both students and buses not to wait so long, therefore, low-carbon transport from buses, where low-carbon public transport leads to green transportation which is essential to carry out green city initiatives [3].

In this paper, the objective is to predict the arrival time of the IUL shuttle buses to Wardanieh campus, as well as monitor the position, speed, humidity and temperature of the buses. Therefore, in order to estimate the arrival time of the bus to the Wardanieh campus, the factors we chose to affect the output time were: the day, the location including latitude and longitude, and the traffic status.

The remaining organization of this paper is as follows: In Section II, we introduce some related works. In Section III, the monitoring system is proposed. Section IV presents the prediction method and data analysis. In section V, we show and discuss the simulation results. At last, in Section VI, the paper is concluded.

2 Literature Review

During the past decade, numerous bus monitoring and tracking systems are investigated [4] - [13]. Furthermore, different arrival time estimation systems based on neural network are studied [15] - [17].

The authors of [4], designed an access point for passenger information system available on public displays at bus stops and terminals based on GSM technology. This system provides passengers with information about the arrival times and location of their buses. A university real-time bus tracking system through a mobile application and GPS module is proposed by authors of [5], where the real-time location of a bus is provided on a user interface screen to the remote users. Another similar project based on mobile application accompanied with a GPS Tracker and IoT people counter devices based on GSM/cellular

network is implemented in [6] to estimate the arrival time as well as the number of people in the bus. A smart tracking system for school buses using passive radio frequency identification (RFID) technology to enhance child safety is implemented in [7]. This system is based on the combination of RFID, GPS, and General Packet Radio Service (GPRS) technologies. In [8], a vehicle tracking system using the Global Positioning System (GPS) and Global System for Mobile Communications (GSM) modem is proposed, it provides users with the capability to track vehicle remotely through the mobile network. A bus monitoring system by using RFID and integrated sensing technologies such as GPS, GPRS and geographic information system (GIS) is proposed in [9], and used to monitor the movement of a bus. Moreover, in [10] and [11] smart bus tracking systems based on RFID are proposed. A public bus tracking system based on GSM/GPRS technologies and using GPS is implemented in [12], where it provides the count of number of passengers in bus and also the estimated arrival time to the user. Moreover, in [13], a vehicle monitoring system (VMS) based on LoRa protocol is developed to monitor different environmental parameters, including ambient temperature and humidity, and air quality parameters, such as PM2.5, NO2, CO, and O3. In order to Collect Lebanon roads downtime events from social media, a real-time online Lebanese road accident platform is proposed in [14].

An attempt has been made by the system proposed in [15] to develop a travel time prediction model for urban bus route in Delhi, India with the help of artificial neural network. Estimation of arrival time based on deep neural network is discussed in [16], where the model can effectively capture the spatial and temporal dependencies in the given path at the same time. Another similar project is the bus arrival time prediction Using RBF neural networks Adjusted by online data is implemented in [17].

To the best of our Knowledge, any complete bus tracking and monitoring system with arrival time estimation based on LoRa technology and RBF neural network is not implemented and studied yet.

3 Bus Monitoring System

3.1 System Topology

The proposed system tracks and monitors the temperature, humidity and speed of a university bus using LoRa technology. This system is based on an offline topology, where the server is located in the same Local Area Network (LAN) with the gateway, which is integrated on a Raspberry pi as shown in Fig. 1. After tracking the bus's location, the system has the capability to estimate, through the software, the remaining time for the bus arrival to a specified destination. Generally, this system mainly operates with temperature and humidity sensors, a speed sensor and a GPS module that are attached to the bus. Firstly, the GPS receives the satellite signals, where the position coordinates include the longitude and latitude. Then, the measured parameters, as well as the data measured by the GPS module, will be processed by a controller that will send the data to a LoRa module. It's important to mention here that the LoRa module in addition to the controller (mainly Arduino uno) form the LoRa node that sends the data to the gateway via LoRaWAN protocol. In our project, we used two shields: LoRa shield and the LoRa/GPS shield. After the data is received by the gateway, it is uploaded to the backend application. To illustrate, data is received by the gateway which is attached to the Raspberry pi, so the Raspberry pi (accompanied with a raspbian image) will process the data and send all the measured information to the local server via an Ethernet network. The data is stored in the database, then using the parameters measured by the GPS module, the application will be able to track the bus and calculate the remaining time for arrival, based on neural network principle. As a final step, the data records are displayed on user friendly interfaces, server application and android application.

3.2 System architecture

The system architecture contains three main modules as presented in Fig. 2, the end-node, the gateway with the control unit and the server application and the android based mobile application. Where the first module, the end-node, contains the temperature, humidity and speed sensors, which are connected to an Arduino micro-controller having a LoRa/GPS shield interface module. The second module is the LoRa Gateway connected to the control unit, which is a Raspberry pi. The components used to implement the proposed system are listed below:

- **LoRa Gateway:** It is a device that connects different types of networks and used to bridge the LoRa wireless network along with the IP network. Mainly, it processes the data then sends it from a LoRa device to the server through an IP based network. We have chose the iC880A-SPI LoRa concentrator as a LoRa gateway.
- **Microcontroller board:** Arduino Uno is one of the best types of Arduino devices. Arduino is an open-source electronic platform based on user friendly hardware and software.
- **LoRa Shield:** It is a LoRa wireless module added to an Arduino Uno board in order to build a LoRa sensor node.
- **LoRa GPS Shield:** Used to add a GPS and a LoRa wireless module to an Arduino board, to build the LoRa sensors, it provides also geographical info. In the GPS part, the used GPS is L80, which is designed mainly for applications that need a GPS connected via the serial ports to the Arduino microcontroller.
- **Sensors:** Here we used the DHT11 basic temperature and humidity sensor, which is considered as a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air.
- **Vehicle Speed Sensor (VSS):** The functionality of this sensor depends mainly on the generation of a magnetic pulse, in the form of a wave, proportional to the speed of the vehicle. This sensor will give the bus speed so we can know the traffic status.
- **Single Board Computer (SBC):** We used a Raspberry pi 3, which has lots of features as well as a terrific processing speed, making it suitable for advanced applications. In our case, the raspberry pi will process the data received from the LoRa gateway and send these data to the application server.

3.3 LoRa Coverage

Before any implementation, it is essential to choose the right location to place the LoRa gateways in order to communicate with LoRa nodes and receive the data all the way from Khaldeh campus to Wardanieh campus. Several Wi-Fi and radio frequency (rf) planning software tools and simulation Platforms already exist [21], [22]. For the simulation, we used HTZ communication software which is the ideal tool for planning homogeneous and heterogeneous wireless networks. During the simulation, two scenarios took

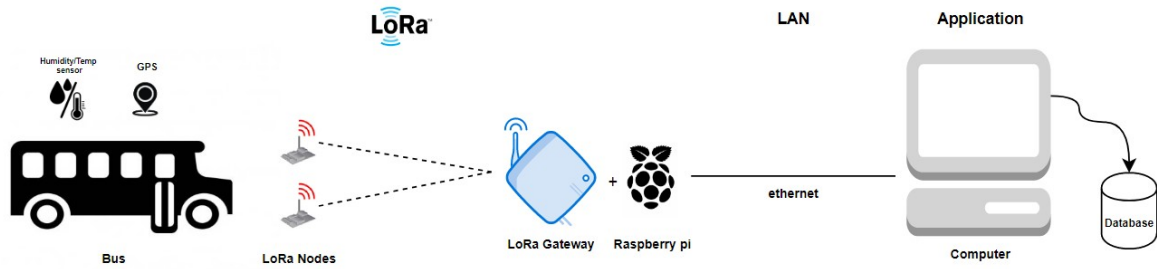


Figure 1: Bus Monitoring System Topology.

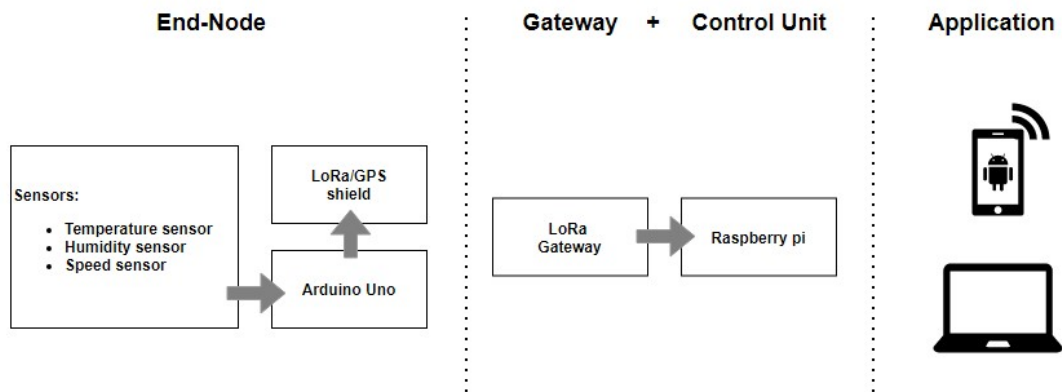


Figure 2: Bus Monitoring System Architecture.

place. In the first scenario, we tried to put the LoRa gateway in the Wardanieh campus but the coverage was not enough due to all the terrains and mountains, so the line of sight is an important factor for LoRa to give a good coverage. In the second simulation, we placed the gateway closer to the main road and the results were great. It's important to mention here that the simulation took almost 45 minutes and we worked on the parameters listed in table 2.

In simulation, we placed two gateways, one in Wardanieh (latitude = 33.603078 and longitude = 35.395546) and another one in Khaldeh campus (latitude = 33.782072 and longitude = 35.473078), both using the same configuration mentioned in table 2. After the simulation, we figured the best location to place the gateway at, in order to have the full coverage for the road between the campuses as shown in Fig. 3.

4 Prediction Method and Data Analysis

In this work, the goal is to estimate the arrival time of the bus for IUL from Khaldeh campus to Wardanieh

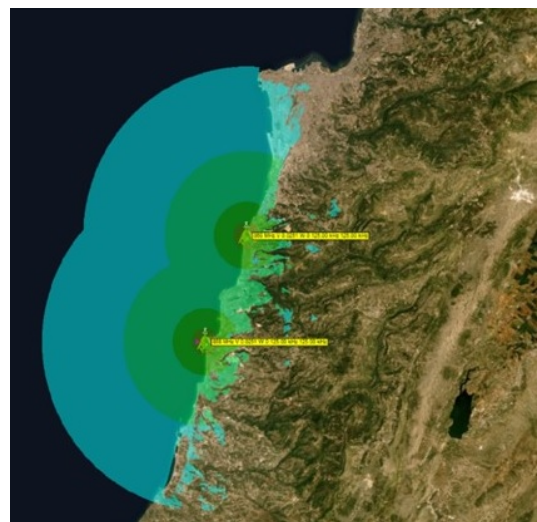


Figure 3: LoRa coverage simulation.

Table 2: Parameters used in the simulation

Frequency	Height	Bandwidth	Average Power	Spreading Factor
868 MHz	35 meters	125 KHz	0.0251 watt	12

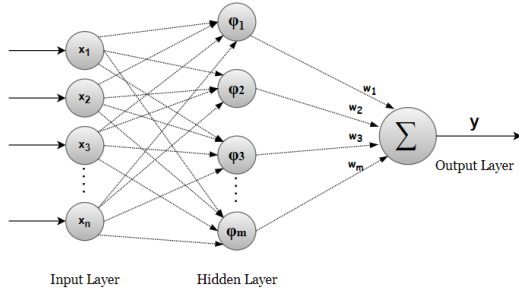


Figure 4: Radial Basis Function Neural Network.

campus, using neural network. The neural network systems usually have different types that perform several functions including classification and prediction. In our case, to find the arrival time, the system needs to predict the output value. Therefore, we decided to implement our system using the RBF Neural Network (RBFNN), which is based on approximation and prediction.

4.1 RBF Neural Network

RBFNN is one of the most popular types of artificial neural networks due to a number of advantages compared to the other types. These advantages include better approximation capabilities, simple network structures and faster learning algorithms [18].

RBFNN, is a multi-layer feed-forward network accompanied with a single hidden layer nonlinear processing units, as well as an output layer of linear weights. A transfer function in the hidden unit layer having radial symmetry with respect to a center characterizes such a network [19]. The radial basis function is generally a Gaussian function. Upon the specification of the radial basis functions, the only adjustable parameters of the network are the weights. These weights can be adjusted using a straightforward least squares approach, since they are linearly related to the output y and hence the output error. This has made radial basis networks attractive in identification and control. The basic architecture of an RBFNN is a 3-layer network as shown in Fig. 4.

The RBF neural network consists of a layer performing linear or non-linear functions of the given attributes, followed by a layer of weighted connections to nodes whose outputs have the same form as the target vectors. This network has a structure like the multi-layer perception neural network with one

hidden layer, except that each node of the the hidden layer computes an arbitrary function of the inputs using Gaussian. Whereas the transfer function of each output node is the trivial identity function. Any function used in the hidden layer has its own parameters, for example, Gaussian widths and positions [20]. The output of the net is given by the following expression:

$$y(x) = \sum_{j=1}^m w_j \phi_j(X), \quad (1)$$

where $\phi = \{\phi_j : j = 1, \dots, m\}$ are the basis functions set, w_j is the associate weight for every radial basis function and the input X is a vector of real numbers. The basis function ϕ can be calculated as a Gaussian function using the expression

$$\phi(x) = \exp\left(-\frac{(X - c)^2}{r^2}\right), \quad (2)$$

Where c is the central point of the function and r is the radius.

4.2 Dataset

To carry out with RBFNN, real data must be gathered to ensure reliability and accuracy of the time predicted. To gather the data, first we need to choose the factors that will affect the output and use them as inputs of the neural network. In our case, the factors were: days of the week, latitude, longitude and the traffic status. We gathered 1500 data using the provided information by google map along the week days, from Monday to Friday. Concerning the longitude and latitude, we chose 100 points between Khaldeh and Wardanieh campuses. With respect to the traffic, the data measured will be the speed, which will be approximated good, bad or very bad.

4.3 Normalizing Data

In order to change numeric columns values in the dataset to utilize a typical and common scale without distorting differences in the values ranges or losing data, the normalization is done. By normalizing, the units of measurements for data are eliminated, then comparing data from different types and sources is possible. Re-scaling data to have values between 0 and 1 is one of the most well-known method for normalization, which is called feature scaling and the formula used is expressed as

$$X_{\text{new}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}. \quad (3)$$

Table 3: MSE for different cases of data division

Case Number	Training	Testing	MSE
1	50%	50%	0.25%
2	75%	25%	0.22%
3	80%	20%	0.18%

4.4 Implementation and Results

First, we started with the basics where the system needs to be trained and tested. We tested three cases on how the data must be divided between training and testing: 50 % training and 50% testing, 75% training and 25% testing, 80% training and 20% testing. Table 3 presents the mean square error (MSE) for the different cases of data division.

Since the error resulting in the case of 80% learning and 20% testing is the lowest, we decided to complete our implementation based on this case. Thus, the training dataset will be a matrix of 1200 records in 4 columns presenting consecutively the day, the latitude, the longitude and the traffic, where the desired output Y is the time of arrival. As a result of using the IBM SPSS software, the used training format of the regression function can be expressed as:

$$\begin{aligned}
 Y = & 0.072407 + 0.060394 \times \text{Day} \\
 & + 0.847760 \times \text{latitude} - 0.069825 \times \text{longitude} \\
 & + 0.044318 \times \text{traffic}
 \end{aligned}
 \tag{4}$$

In our work, we used the Matlab newrb function to create the RBFNN which has three parameters: the input, the target and the spread. By increasing the spread, the function approximation is more smoother. If the spread is too large, then lot of neurons are needed in order to fit a fast-changing function. Moreover, if the spread is too small, then in order to fit a smooth function, many neurons are needed, and the proposed network might not generalize well. Therefore, to find the best spread value for any problem, the newrb function has to be called with different spread values. Fig. 5 illustrates a graph showing the change of error with respect to different values of spread. It's obvious here that the error stopped changing at almost spread equal to 0.1 with the error being equal to 0.18%. Table 4 shows the results of RBFNN by using the different factors. The results are close to the real results, and the mean square error is equal to 0.18%.

5 Software Realization

Finally, The data measured as well as the estimated time of arrival are displayed on a user friendly software, to permit the administrator to manage and monitor the buses. This software includes a graphical

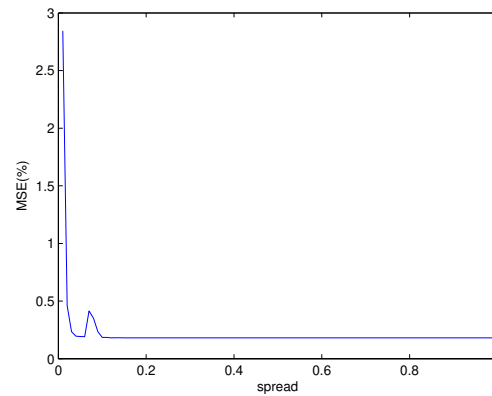


Figure 5: MSE with respect to spread.

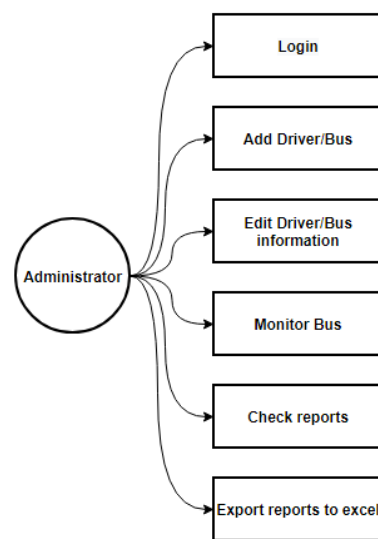


Figure 6: System requirements.

user interface, implemented in Java, linked to an SQL Server database.

Fig. 6 shows all the software system privileges that are implemented. These privileges are: accessing the system, adding, updating and deleting drivers/bus information and monitoring the buses as checking the estimated time of arrival, location, distance to arrive, speed, temperature and humidity. Furthermore, the administrator can check reports and export them to an excel sheet. Moreover, the exact location and the arrival time are displayed also on a mobile android-based application for IUL students. It's a very simple application were any IUL student just needs to enter his id and login. After logging in the students can press the monitor button to display the information. Moreover, when pressing on show map every student will be able to find the exact location of the bus at a certain time as shown in fig. 7.

Table 4: Results of testing in MATLAB

Number	Day of the week	Latitude	Longitude	Traffic status	Predicted Arrival time	Real Arrival Time
1	Monday	33.645172	35.401281	Very bad	7.64133 mins	8 mins
2	Wednesday	33.726706	35.452786	Very bad	15.70433 mins	16 mins
3	Monday	33.780239	35.474503	Bad	20.9915 mins	21 mins
4	Tuesday	33.63035	35.402069	Good	3.89991 mins	4 mins
5	Thursday	33.691489	35.425592	Good	8.979113 mins	9 mins
6	Friday	33.776497	35.472042	Bad	18.6799 mins	19 mins

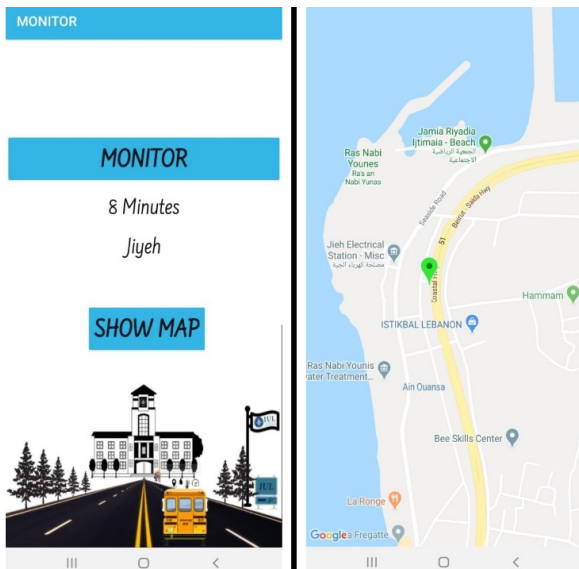


Figure 7: Monitoring mobile application results.

6 Conclusion

The bus monitoring system proposed in this paper could not only detect the bus's location, but also collects temperature/humidity/speed information and estimates the arrival time using the neural network (RBFNN) where testing results show that the MSE is small. All of the data are gathered and presented in a graphical user interface for the administrator to display the information in addition to an android application for students to check the bus location. It's important to mention here that this work is not limited; lots of other services and improvements can be added. Finally, the proposed system accompanied with the android application will completely help in reducing the discomfort of the students as well as promoting their safety.

References:

[1] P. Yadav and S. Vishwakarma, "Application of Internet of Things and Big Data towards a Smart City," *In proc. IEEE International Conference*

On Internet of Things: Smart Innovation and Usages (IoT-SIU), 2018, pp. 1-5.

[2] W. Ayoub, A. E. Samhat, F. Nouvel, et al., "Internet of Mobile Things: Overview of LoRaWAN, DASH7, and NB-IoT in LPWANs Standards and Supported Mobility," *IEEE Communications Surveys & Tutorials*, Vol. 21, No. 2, 2019, pp. 1561-1581.

[3] Putrajaya Holdings, "Initiative Report Green City Low Carbon Green City," *Putrajaya*, 2012, pp. 37-40.

[4] K. V. Padmaja, N. Jeevagan, P. Santosh, et al., "Project track my bus," *In proc. IEEE India Conference (INDICON)*, 2014, pp. 1-4.

[5] D. Patel, R. Seth, V. Mishra, et al., "Real-Time Bus Tracking System," *International Journal of Advanced Research in Computer Engineering and Technology (IJARCET)*, Vol. 4, No. 3, 2015, pp. 743-746.

[6] S.A. Sharif, M.S. Suhaimi, N.N. Jamal, et al., "Real-Time Campus University Bus Tracking Mobile Application," *In proc. Seventh ICT International Student Project Conference (ICT-ISPC)*, 2018, pp. 1-6.

[7] K. Shaaban, A. Bakkali, E. B. Hamida, et al., "Smart tracking system for school buses using passive RFID technology to enhance child safety," *Journal of Traffic and Logistics Engineering*, Vol. 1, No. 2, 2013, pp. 191-196.

[8] S. P. Metkar and G. L. Deshmukh, "RTOS based vehicle tracking system," *In proc. IEEE International Conference on Information Processing (ICIP)*, 2015, pp. 621-624.

[9] M. A. Hannan, A. M. Mustapha, A. Hussain, et al., "Intelligent bus monitoring and management system," *In proc. of the World Congress on Engineering and Computer Science (WCECS)*, Vol. 2, 2012, pp. 24-26.

- [10] P. Godge, K. Gore, A. Gore, et al., "Smart bus management and tracking system," *International Journal of Latest Engineering Science*, Vol. 2, No. 2, 2019, pp. 20-25.
- [11] V.M. Anu, D. Sarikha, G.S. Keerthy, et al., "An RFID based system for bus location tracking and display," *In proc. International Conference on Innovation Information in Computing Technologies*, 2015, pp. 1-6.
- [12] A. J. Kadam, V. Patil, K. Kaith, et al., "Developing a Smart Bus for Smart City using IOT Technology," *In proc. IEEE International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018, pp. 1138-1143.
- [13] C. L. Hsieh, Z. W. Ye, C. K. Huang, et al., "A vehicle monitoring system based on the LoRa technique," *International Journal of Transport and Vehicle Engineering*, Vol. 11, No. 5, 2017, pp. 1100-1106.
- [14] A. Ghandour, H. Hammoud, M. Dimassi, et al. "Allometric scaling of road accidents using social media crowd-sourced data," *Physica A: Statistical Mechanics and its Applications*, vol. 545, 2020.
- [15] J. Amita, J. S. Singh and G. P. Kumar, "Prediction of bus travel time using artificial neural network," *International Journal for Traffic and Transport Engineering*, Vol. 5, No. 4, 2015, pp. 410-424.
- [16] D. Wang, J. Zhang, W. Cao, et al. "When will you arrive? estimating travel time based on deep neural networks," *In proc. Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [17] L. Wang, Z. Zuo and J. Fu, "Bus arrival time prediction using RBF neural networks adjusted by online data," *Procedia-Social and Behavioral Sciences*, Vol. 138, 2014, pp. 67-75.
- [18] H. Sarimveis, A. Alexandridis, S. Mazarakis, et al. "A new algorithm for developing dynamic radial basis function neural network models based on genetic algorithms," *In proc. Elsevier Computer Aided Chemical Engineering*, Vol. 10, 2002, pp. 949-954.
- [19] M. C. Mackey and L. Glass, "Oscillation and chaos in physiological control systems," *Science*, Vol. 197, No. 4300, 1977, pp. 287-289.
- [20] P. G. Jayant, V. H. Tatwawadib, S. G. Mahakalkarc, et al., "Comparative analysis of Multilayer Perceptron and Radial Basis Function ANN for prediction of cycle time of Structural subassembly Manufacturing," *International Journal of Science, Spirituality, Business and Technology (IJSSBT)*, Vol. 3, No. 1, 2014.
- [21] A. Ghandour, J. Haydar, A. Hariri, et al. "Aparecium Wi-Fi Planner using Enhanced Indoor Propagation Model," *In proc. IEEE International Wireless Communications and Mobile Computing (IWCMC)*, 2020.
- [22] J. Haydar, A. Ibrahim, A. E. Samhat, et al. "ABCDecision: A simulation platform for access selection algorithms in heterogeneous wireless networks," *EURASIP Journal on Wireless Communications and Networking*, 2010.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0

https://creativecommons.org/licenses/by/4.0/deed.en_US